Nos. 14-1516, -1530

United States Court of Appeals for the Federal Circuit

SYNOPSYS, INC.,

Appellant,

– v. –

MENTOR GRAPHICS CORPORATION,

Cross-Appellant,

– v. –

MICHELLE K. LEE, DEPUTY DIRECTOR,
U.S. PATENT & TRADEMARK OFFICE,

Intervenor.

ON APPEAL FROM THE PATENT AND TRADEMARK OFFICE,
PATENT TRIAL AND APPEAL BOARD, CASE NO. IPR2012-00042.

OPENING BRIEF AND ADDENDUM OF APPELLANT SYNOPSYS, INC.

I. Neel Chatterjee
George L. Kanabe
Travis Jensen
Cam T. Phan
Orrick, Herrington & Sutcliffe LLP
1000 Marsh Road
Menlo Park, CA 94025-1015

William H. Wright
Orrick, Herrington & Sutcliffe LLP
777 S. Figueroa Street
Los Angeles, CA 90017-5855

Eric A. Shumsky
Orrick, Herrington & Sutcliffe LLP
Columbia Center
1152 15th Street, NW
Washington, DC 20005-1706
(202) 339-8400
eshumsky@orrick.com

Andrew D. Silverman
Orrick, Herrington & Sutcliffe LLP
51 W. 52nd Street
New York, NY 10019-6142

Attorneys for Appellant Synopsys, Inc.

Claims 1 and 28

1. A method comprising the steps of:

   a) identifying at least one statement within a register transfer level (RTL) synthesizable source code; and

   b) synthesizing the source code into a gate-level netlist including at least one instrumentation signal, wherein the instrumentation signal is indicative of an execution status of the at least one statement.

28. A storage medium having stored therein processor executable instructions for generating a gate-level design from a register transfer level (RTL) synthesizable source code, wherein when executed the instructions enable the processor to synthesize the source code into a gate-level netlist including at least one instrumentation signal, wherein the instrumentation signal is indicative of an execution status of at least one synthesizable statement of the source code.

A126(15:2-8), A127(17:61-18:7) (’376 Patent).

CERTIFICATE OF INTEREST

Counsel for Appellant Synopsys, Inc. certifies the following:

1. We represent Synopsys, Inc.

2. The name of the real party in interest (if the party named in the caption is not the real party in interest) represented: N/A

3. All parent corporations and any publicly held companies that own 10 percent or more of the stock of the party or amicus curiae represented: None.

4. The names of all law firms and the partners or associates that appeared for party or amicus now represented in trial court or agency or are expected to appear in this court are:

ORRICK, HERRINGTON & SUTCLIFFE LLP
Adya Baker
I. Neel Chatterjee
Travis Jensen
George L. Kanabe
Robert M. Loeb
Cam T. Phan
Eric A. Shumsky
Andrew D. Silverman
William H. Wright

SIDLEY AUSTIN LLP
C. Frederick Beckner III
Peter D. Keisler
Erika Myers
M. Patricia Thayer

Date: October 3, 2014

/s/ Eric A. Shumsky

Eric A. Shumsky
Orrick, Herrington & Sutcliffe LLP
Columbia Center
1152 15th Street, NW
Washington, DC 20005-1706
Telephone: (202) 339-8464
Facsimile: (202) 339-8500
eshumsky@orrick.com

Counsel for Appellant Synopsys, Inc.

TABLE OF CONTENTS

Page

TABLE OF AUTHORITIES

Page(s)

FEDERAL CASES

FEDERAL STATUTES

FEDERAL RULES AND REGULATIONS

LEGISLATIVE MATERIALS

FEDERAL CASE MATERIALS

OTHER AUTHORITIES

TABLE OF ABBREVIATIONS

Gregory '109 or "Gregory"	U.S. Patent No. 6,132,109, assigned to Synopsys, Inc.
'376 patent	U.S. Patent No. 6,240,376, the patent-at-issue, assigned to Mentor Graphics Corporation
APA	Administrative Procedure Act, 5 U.S.C. §§ 551 *et seq.*
the Board	Patent Trial and Appeal Board
EVE, S.A.	Emulation and Verification Engineering, S.A.
HDL	Hardware Description Language
Mentor	Cross-Appellant Mentor Graphics Corporation
PTO	Patent & Trademark Office
RTL	Register Transfer Level
VHDL	Very High Speed Integrated Circuit Hardware Description Language

STATEMENT OF RELATED CASES

This appeal arises out of a final written decision of the Patent Trial and Appeal Board ("the Board") that granted in part and denied in part Synopsys, Inc.'s petition for *inter partes* review of U.S. Patent No. 6,240,376 ("the '376 patent"). That patent is assigned to Mentor Graphics Corporation. The following pending matters may affect or be affected by this Court's decision. *See* Fed. Cir. R. 47.5.

After the Board instituted proceedings, Mentor sued the Patent & Trademark Office ("PTO") in the Eastern District of Virginia under the Administrative Procedure Act ("APA"), challenging the Board's decision to institute *inter partes* review. *Mentor Graphics Corp. v. Lee*, No. 1:13-cv-00518 CMH-TCB (E.D. Va.). The district court granted motions to dismiss filed by the PTO and Synopsys (which had intervened). Mentor appealed the dismissal of its lawsuit to this Court, and that appeal has been stayed pending resolution of *Versata Development Group, Inc. v. Lee*, Fed. Cir. No. 2014-1145. *See* Order, *Mentor Graphics Corp. v. Lee*, No. 13-1669 (Fed. Cir. July 30, 2014), ECF No. 37.

There also is pending district court litigation between the parties. In 2010 and again in 2012, Mentor filed suit in the District of Oregon

against EVE-USA Inc. and Emulation and Verification Engineering, S.A. ("EVE, S.A."), companies that later were acquired by Synopsys. *Mentor Graphics Corp. v. EVE-USA, Inc.*, Nos. 3:10-cv-954-MO (LEAD), 3:12-cv-1500-MO (D. Or.). That litigation involves different patents than the one at issue here.

In 2012, Synopsys, EVE, S.A., and EVE-USA Inc. sued Mentor for a declaratory judgment of non-infringement and invalidity in the Northern District of California in a case involving the '376 patent among others. Mentor counter-claimed for infringement. That case was transferred and consolidated with other cases pending in the District of Oregon. *Synopsys, Inc. v. Mentor Graphics Corp.*, No. 3:13-cv-579-MO (D. Or.). In that action, which remains pending, Synopsys and the EVE entities also alleged that Mentor infringes several Synopsys patents, including the patent that is the relevant prior art in this appeal: U.S. Patent No. 6,132,109 ("'109 Patent" or "Gregory").

Finally, Synopsys has filed an APA suit in the Eastern District of Virginia. In it, Synopsys challenges the Board's regulation, and its policy and practice, of failing to "issue a final written decision with respect to the patentability of *any* patent claim challenged by the

petitioner," 35 U.S.C. § 318(a) (emphasis added), as required by statute, and instead issuing final written decisions that address only certain of the challenged claims. The government has moved to dismiss the complaint; that motion remains pending. Mentor has intervened in the case. *See Synopsys, Inc. v. Lee*, No. 1:14-cv-00674-JCC-IDD (E.D. Va.).

INTRODUCTION

When Congress enacted the America Invents Act, it created new post-grant review proceedings to combat the "growing sense that questionable patents are too easily obtained and are too difficult to challenge." H.R. Rep. No. 112-98, pt. 1, at 39 (2011). Appellee Mentor Graphics owns one such questionable patent (No. 6,240,376), which it has been wielding in litigation, *supra* at xiii-xiv—notwithstanding that that patent came after and is anticipated by one of Synopsys' own patents. Accordingly, Synopsys availed itself of the new *inter partes* review procedure. 35 U.S.C. §§ 311-319. The Board initiated review because it found "a reasonable likelihood that … at least 1 of the claims" was invalid. *Id.* § 314(a). Ultimately, the Board cancelled three claims (which are the subject of Mentor's cross-appeal).

The Board, however, did not go quite far enough. Two additional claims of the '376 patent (1 and 28) also are anticipated by the same prior-art reference (Gregory). Gregory and the '376 patent both teach a method of modifying source code in order to make it easier to correct errors in the design of a circuit. As to both challenged claims, the Board properly determined that Gregory discloses every limitation save one.

But, as to that purportedly missing limitation, the Board committed a number of errors—applying the wrong test for anticipation; and failing to address (or sometimes even to consider) arguments made by Synopsys. That aspect of the Board's decision therefore must be vacated.

Separately, the Board erred as a matter of law when—as has become its practice—it issued a final written decision that did not address all of the challenged claims. Not only does this practice contravene the statute's plain text; perhaps not coincidentally, it effectively shields those claims from appellate review. This portion of the Board's decision must be vacated and remanded with instructions to the Board to issue a final written decision that complies with the clear statutory requirement.

STATEMENT OF JURISDICTION

Synopsys petitioned for *inter partes* review of claims 1-15 and 20-33 of the '376 patent. A135-99; *see* 35 U.S.C. § 311. The Board instituted proceedings as to claims 1-9, 11, and 28-29. A1-41. On February 19, 2014, the Board cancelled claims 5, 8, and 9. A42-94. On April 22, 2014, Synopsys timely filed its notice of appeal. A849-55; 35 U.S.C.

§§ 141(c), 319; 37 C.F.R. § 90.3(a)(1). On May 2, 2014, Mentor filed its own notice of appeal. A856-62. This Court has jurisdiction. 28 U.S.C. § 1295(a)(4)(A).

STATEMENT OF ISSUES

1. Both the '376 patent and the prior-art Gregory patent disclose technology for finding errors in the design of a circuit. Specifically, both disclose a method of modifying a circuit's source code to produce an output that indicates whether a particular portion of the code is active, thereby making it easier to find and correct errors in the circuit's design.

Did the Board err in concluding that claims 1 and 28 of the '376 patent are not anticipated by Gregory?

2. The Board institutes *inter partes* review when "there is a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the claims challenged in the petition." 35 U.S.C. § 314(a). "If an inter partes review is instituted, … the Board *shall* issue a final written decision with respect to the patentability of *any patent claim challenged by the petitioner*." 35 U.S.C. § 318(a) (emphases added).

Did the Board err by instituting review but failing to issue a final written decision with respect to the patentability of each claim challenged by Synopsys?

STATEMENT OF THE CASE

In the modern, wired economy, electronic circuits are everywhere. Tiny integrated circuits such as microchips are ubiquitous in computers, cell phones, microwave ovens, pacemakers—indeed, in virtually all electronic equipment. At the simplest level, a microchip is made up of tiny electronic components (like transistors, resistors, and capacitors) that are packed onto a fingernail-sized plate or chip typically made of silicon. *See* Frank Vahid, *Digital Design, with RTL Design, VHDL, and Verilog* 39-40 (2d ed. 2011). Among other things, these chips can process information and serve as an electronic device's internal memory.

As electronic devices increase in complexity, microchips must get smaller and smaller, while doing more and more. It's well known that the number of transistors packed into a microchip doubles approxi-

mately every two years.[1] The very first microchip in the 1950s contained one transistor, three resistors, and one capacitor; today, the average microchip will contain *several hundred million* transistors and *miles* of wires.[2] As microchips and other circuits have become smaller and more sophisticated, it has become especially important to ensure that circuits contain no design flaws (or "bugs"). That verification task once was performed by manually testing circuits. Today, however, problems are identified, and the chip's design is tested, by advanced verification software ("simulators") and hardware ("emulators"). When bugs are identified, humans can go back to "debug" the design. This case involves technology for tracing bugs found during verification back to the original source code that was used to create the circuit design, thereby making it easier for humans to find and fix the problems in the code.

---

[1] *See* Gordon E. Moore, *Cramming More Components Onto Integrated Circuits*, Electronics, April 19, 1965 at 114, *available at* http://tinyurl.com/lrj5sgg.

[2] *See* Xinghao Chen & Nur A. Touba, *Electronic Design Automation: Synthesis, Verification, and Test* ch. 2, 39 (Laung-Terng Wang et al., eds. 2009).

Before describing the invention claimed by Mentor, and the Synopsys patent that preceded it, we begin with a brief background about the process of designing microchips.

I. DESIGNING INTEGRATED CIRCUITS AND VERIFYING THAT THEY OPERATE AS PLANNED

A. Source Code Description

"[T]he design of a[] … chip[] can be an extremely complicated task." A1596. Even a decade ago, it was not uncommon for a single chip to have more than a million components. Because that level of complexity goes far beyond what a human can process, modern chip design typically begins with a high-level "description of what the chip is supposed to do," called a specification. *Id.*

The next step is to begin designing the chip itself, using a "formal language[] similar to [a computer] programming language[]." *Id.* Such languages are known generally as hardware description languages (HDLs), two of the most common of which are Verilog and VHDL (Very High Speed Integrated Circuit Hardware Description Language). *Id.*; A119(1:16-31) ('376 patent). By way of example, here is a small snippet of HDL code written in VHDL:

```
if (C and B) then
        Z <= not(A or B);
else
        Z <= not B;
end if;
```

A1108 Fig. 4 (Gregory '109). This code describes how the circuit should behave, using a common "if-then-else" statement: "[I]f" one condition is satisfied, "then" follow one branch; otherwise ("else"), follow a different branch. Inputs in a digital circuit are always equal to either 1 or 0, which are sometimes referred to as "high" or "true" (1), and "low" or "false" (0). That being so, this sample code provides a simple instruction:

- "If" inputs "(C and B)" both are true (i.e., both equal to 1), "then" the output ("Z") should be the opposite of inputs A or B ("not(A or B)").
- Otherwise ("else," which is to say if inputs "(C and B)" are *not* both equal to 1), the output ("Z") is the opposite of input B ("not B").

A1897 ¶ 57(Mentor's Expert's Decl.).

The output, Z, is the electronic value, or "signal," generated by the circuit. Like inputs, outputs equal either 1 or 0. And, because an output is the product of the logical operations that a circuit performs on

one or more inputs, an output's value can convey information about the circuit that produced it.

Millions upon millions of logical operations like this one are built into the design of a microchip. And, as described below, this functional description ultimately is translated into physical circuitry and carried out using transistors and other components that control the flow of electrons through the chip.

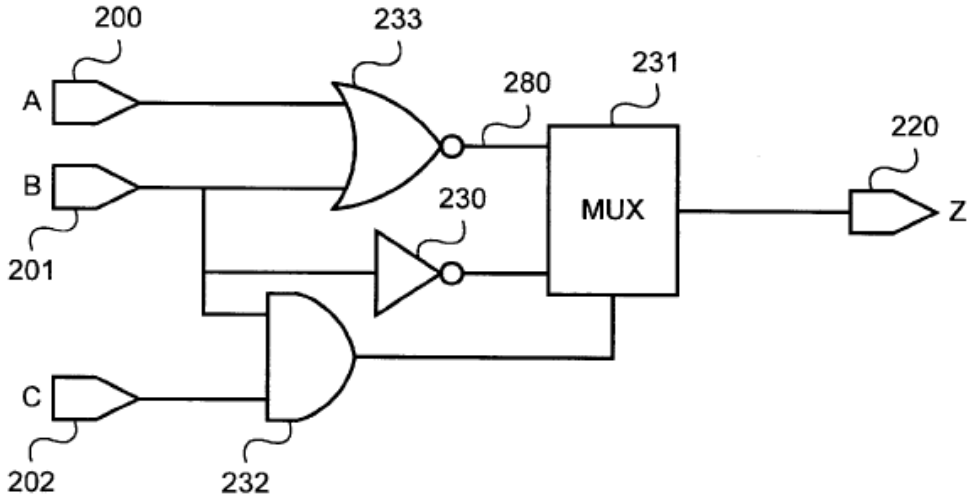
After the designer has described a desired function in HDL, it is common to "simulate" the circuit using a computer program in order "to determine whether the circuit produces correct values in response to inputs." A1139(4:41-52) (Gregory '109). As discussed below (at 15-16), simulation is repeated throughout the process of designing and fabricating circuits. If there is a problem, the designer can fix it by modifying the code. A1139(4:53-57). If not, the designer moves to the next step: logic synthesis.

B. Logic Synthesis

The HDL code description is fed into a computer program that performs logic synthesis. First, a translator program takes a subset of the HDL code (which it recognizes as register transfer level (RTL)

source code)—like the snippet depicted above—and "synthesize[s]" or "translat[es]" it into a "gate-level netlist." A1140(5:4-8) (Gregory '109); A119(1:35-36) ('376 patent). This netlist is a text file that provides a textual description of the circuit, including components such as "logic gates" (more on those below, at 10-12) and the interconnections between them. Then, the program can "convert[]" the netlist into a schematic view of the circuit. A1138-39(2:59-3:3) (Gregory '109). In architectural terms, the netlist is akin to a detailed, textual description of the plan for a building, while the schematic is the blueprint. They simply provide the same information in different ways, and both the netlist and schematic "correspond directly with the statements in the source HDL." A1140(5:4-8) (Gregory '109).

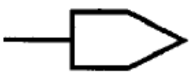
For instance, the snippet of HDL code depicted above (at 7) would be synthesized into a gate-level netlist (describing the interconnection of various logic gates), then converted into the following corresponding schematic:



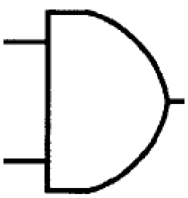
A1110 Fig. 6 (Gregory '109). So, just as the netlist describes a circuit's different components and their interconnections, a schematic graphically depicts those components and the wires connecting them, which together carry out the functions described by the HDL code. In particular, the schematic shows "logic gates" ("gates," for short), like 232 and 233 in this diagram, which are represented as geometric shapes. And although these components resemble hieroglyphics (the Rosetta Stone for which appears below, at 11-12), they simply depict graphically the same types of logical operations described in the code above. They take inputs, perform a logical operation required by the code, and produce an output. A1138(2:63-3:1) (Gregory '109), A1108 Fig. 4 (Gregory '109).

Before describing the next step in circuit design, it's worth pausing to explain the symbols used in a circuit diagram like the one above.

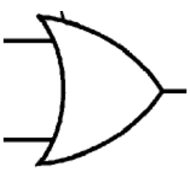
This symbol (200, 201, and 202 in the diagram) represents an input (A, B, and C above).



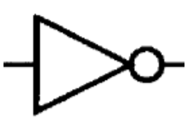
This symbol (220) represents an output (Z above).



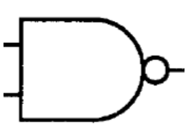
This symbol (232) is an "AND gate." It receives inputs (the lines on the left) and returns an output of 1 ("true") if *all* inputs are 1. Otherwise, the output will be 0. The "if" statement in the code depicted above ("if (C and B) then …"), contains an AND statement: That condition is satisfied only if all inputs (both C and B) equal 1.



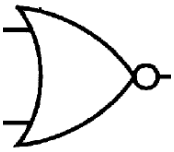
This symbol (which does not appear in the exemplary diagram) is an "OR gate." It returns an output of 1 if *any* input is 1.



This symbol (230) is a "NOT gate" or "inverter." It receives a single input and generates an opposite output: If the input is 1, the output is 0, and vice versa.



This symbol is a "NAND" gate. Its output is the opposite of an "AND" gate. Whereas an AND gate returns a 0 if not all its inputs are 1, a NAND gate returns a 1 if not all of its inputs are 1. Similarly, while an AND gate returns a 1 if all inputs are 1, a NAND gate returns a 0 if all inputs are 1.



This symbol is a "NOR" gate. It produces the opposite output of an OR gate. Thus, while an OR gate returns an output of 1 if any input is 1, a NOR gate returns a 0 if any input is a 1. A NOR gate only produces an output of 1 if all inputs are 0.[3]

See John F. Wakerly, *Digital Design, Principles & Practices* 76-77 (2d ed. 1994).

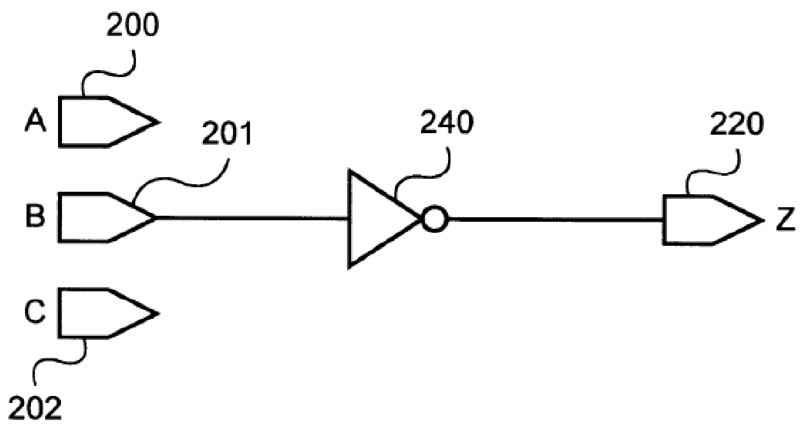
After translation comes "the optimization step of the synthesis process." A1145(16:39-41) (Gregory '109). As noted above, during translation, a computer program translates the HDL code written by a circuit designer into a netlist, which is then converted to a schematic that depicts the circuit graphically. That translation is a literal one: The synthesis program "converts [the designer's] description into gates and other circuit structures that directly correspond statement by statement with the designer's description." A1138-39(2:66-3:1) (Gregory '109).

Often, however, there is more than one way to achieve the designer's intended result—particularly when multiple logical operations

---

[3] The other component, labeled "MUX," is a "multiplexer." This is a circuit component that, based on instructions it receives, selects one of several inputs to provide as the output (much like a railroad switch selects which incoming track to connect to a continuing track).

are being combined—and the designer may not have picked the best one. Thus, after the HDL code has been *translated* into a netlist and gate-level schematic, the circuit is then *optimized*, a process through which software removes unnecessary components from the circuit. Optimization results in a simpler and often more efficient design, which can be simulated (tested) more quickly. For example, this schematic shows an optimized version of the circuit depicted above (at 10) and in Gregory Figure 6:



A1111 Fig. 7 (Gregory ’109). This version of the circuit is much simpler than the one in Figure 6, *cf. supra* at 10, but both are functionally

equivalent to the if-then-else statement in Figure 4. A1142(9:51-53) (Gregory '109), A1143(12:33-42) (Gregory '109).[4]

One potential disadvantage of optimization, however, is the loss of information and components (like inputs and outputs) that may be necessary to pinpoint the source of problems that are identified through the next step: verification.

C. Verification

A "circuit designer needs to ensure that the circuit performs the correct function." A1138(1:21-23) (Gregory '109). Accordingly, after a proposed circuit has been synthesized and optimized, its design will generally be simulated again to ensure that it functions as intended. Testing is done by entering various combinations of inputs to ensure correct outputs. Because "complex designs," such as microchips, often involve "millions or billions" of combinations of test inputs,

---

[4] For the truly initiated, this is because the output (Z) will always be the opposite of B. If "B" is 0, the condition "if (C and B)" necessarily results in 0, and the "else" statement, "Z <= not B" will be executed. If "B" is 1, then it doesn't matter whether "if (C and B)" is 0 or 1, because either Z is "not(A or B)" and the value of B as 1 necessarily results in a value of 0 for Z (the "then" statement) *or* Z is "not B" (the "else" statement). Either way, Z is the opposite of B. Because "Z" is determined independently of "C" and "A," they are irrelevant to the circuit's function and removed through optimization.

A120(4:58-60) ('376 patent), the "[d]ata volumes" of such tests are "beyond any limit that could be overseen [by human beings,]" and thus "powerful tools are needed for verification," A1603.

Two common tools for verifying circuits are software simulators and hardware emulators. In simulation, a computer program predicts the behavior of a circuit through software. A119(1:35-43) ('376 patent). Test inputs are entered, and the resulting outputs are checked against the expected outputs. Because simulations are run using computer programs, they are considered easy, accurate, flexible and low cost.

Emulators are large pieces of advanced, customizable hardware that can imitate the behavior of circuitry in order to test it. The hardware can be reconfigured to incorporate changes to the circuit's design. Emulators are typically much faster than simulators and have the further advantage that they produce physical, functional exemplar circuits. Emulators, however, are more expensive than simulators.[5]

---

[5] Yuichi Nakamura et al., *A Fast Hardware/Software Co-Verification Method for System-On-a-Chip by Using a C/C++ Simulator and FPGA Emulator with Shared Register Communication*, ACM, Proc. of the 41st Ann. Design Automation Conf. 299 (June 2004), *available at* http://tinyurl.com/pppkrlb.

Thus, simulators and emulators serve similar purposes—to test a circuit's design to ensure that it will function as expected. And, if errors are uncovered during verification, the designer can "go back to the HDL [source code description], where the function is specified and make adjustments there" to fix or debug the circuit. A1140(6:26-30) (Gregory '109).

Finally, once the circuit has been verified and debugged, the chip is ready to be manufactured. *Id.*(6:42-43).

* * *

This case involves technology for debugging a circuit design after the design has been synthesized and optimized. In particular, this technology permits a designer who has learned of a bug to locate the source of the problem in the HDL code, where it can be fixed.

II. SYNOPSYS' PRIOR ART AND MENTOR'S '376 PATENT

A. Synopsys Develops A Method To Trace Elements Of A Circuit Back To The HDL Source Code Description.

Synopsys was founded more than 25 years ago, when the internet wasn't what it is today, and integrated circuits weren't pervasive in cars, buildings, and appliances. As consumer and wireless products have proliferated, Synopsys has played an essential role in this

innovation. Among its other advances, Synopsys supplies engineers with software that is used to design and test integrated circuits to make sure that they function as intended.[6]

One of Synopsys' innovations is embodied in a patent applied for by a group of Synopsys inventors (including Brent Gregory) in June 1994. It ultimately issued as U.S. Patent No. 6,132,109 ("Gregory '109" or "Gregory"), and was assigned to Synopsys. A1103-48.[7] Gregory facilitates "debugging digital circuits constructed using logic or behavioral synthesis" by making it easier to trace problems in the circuit design back to their origin in the source code. A1103(Abstract), 1138(1:12-15). As discussed above, when HDL code is synthesized into a netlist and circuit diagram, the resulting draft circuit "is generally large and slow," and so it will be optimized to create "a functionally equivalent, yet improved structure." A1139(3:4-8). But optimization creates a difficulty of its own: Simplifying the circuit modifies it, and

---

[6] *See* Synopsys, Inc., 2013 Annual Form 10-K Report at 2, *available at* http://tinyurl.com/mjcmgal.

[7] Claim 1 of Gregory was recently held indefinite in litigation that remains ongoing. *See* Summary Judgment Dec., *Mentor Graphics Corp. v. EVE-USA, Inc.*, No. 3:10-cv-954 (D. Or. July 25, 2014), ECF No. 581.

thereby eliminates the "direct[] correspond[ence]" between the HDL code and the circuit diagram. A1138(2:67). "Prior to optimization, it is a straight-forward task to identify which circuit element of the initial circuit corresponds to what part of the HDL source code." A1140(5:14-17). But "because of the extensive manipulations performed during the optimization process, such identification after optimization becomes almost impossible." A1140(5:17-20). Thus, if there is an error in the now-modified circuit, it will be difficult to find the corresponding HDL code to "debug[]" it. A1139(3:12-26). This is problematic because "[i]deally, the designer would go back to the HDL [code] where the function is specified and make adjustments there." A1140(6:28-30). If a designer cannot modify the HDL code directly, she must make changes in the netlist or schematic. Because the fixes for those bugs will not be in the HDL code, the code no longer matches the circuit design, and the code cannot simply be resynthesized later when the designer is ready to fabricate the chip. A1139(3:23-32).

Gregory devised a novel solution to combine the benefits of optimization with the benefits of debugging at the HDL code level. A1141(7:64-66). In particular, Gregory "provid[es] a designer with the

ability to mark the synthesis [HDL] source code in the places that the designer wants to be able to debug" by inserting "a particular text phrase, such as 'probe', along with some additional optional information." A1141(8:11-26). For instance, Gregory shows the same sample code set forth above (at 7), modified to contain a "probe statement":

```
if (C and B) then
        Z <= not(A or B);   --Synopsys probe_statement
else
        Z <= not B;
end if;
```

A1112 Fig. 8 (highlighting added).

This modified code (known as "instrumented code," or code containing "instrumentation") sends a signal to the software that is responsible for synthesizing the code into a circuit and then optimizing the circuit. Specifically, the modified code signals to the software that, when the software produces a circuit, it should do so without removing or replacing the circuit components associated with the instrumented code. A1143(11:15-41). If necessary, the synthesizer or optimizer may add circuitry in order to maintain the components associated with the identified source code. A1141(8:14-42). In the language of Gregory, the synthesizer and optimizer should "generate[] a circuit th[at] provides the same function as it did without the 'probe' statement, but add[]

additional information or components to the initial circuit that indicate that certain components should not be replaced during optimization." A1141(8:26-30).

Gregory demonstrates the results of this graphically, by showing the synthesized and optimized circuits that will result from *non*-instrumented code, A1110-11 Figs. 6-7, and (for comparison purposes) the quite different synthesized and optimized circuits that would result from instrumenting the underlying code, A1113-14 Figs. 9-10. As this comparison demonstrates (*infra* at 38-40), the process of instrumentation protects the circuit components associated with the instrumented code. *See* A1143-44(12:33-13:20).

In short, Gregory teaches instrumenting HDL code to identify and mark particular source code, and to preserve or "add[] additional … components to the initial circuit." A1141(8:21-29). This ensures that the "components will not be replaced during optimization," and that, even after optimization, those circuit components will remain "directly and traceably related to those components in the initial circuit." A1141(8:21-42). That these components can be traced back to the

source code in turn "facilitates debugging" at the HDL code level. A1141(8:31-42).[8]

B. Mentor Patents A Method For Marking And Instrumenting Hardware Description Code.

Four years after Synopsys applied for the Gregory patent, an application was filed that resulted in the '376 patent. That patent lists Mentor as the assignee. A95. Like Gregory before it, the '376 patent is aimed at making it easier to debug the HDL code for microchip designs by "instrumenting" the HDL code before it is synthesized. A119(2:25-29) ('376 patent).

The problem the patent sought to address is that during synthesis, "most of the high-level information from the RTL source code is lost," and without that "information, many of the debugging functionalities are unavailable." A119(2:1-4); *accord* A120(4:64-66). During

[8] Gregory discloses that, in addition to protecting a single line of code (as in the example above), the same method can be used to protect blocks of code. Gregory explains that a "block probe" can mark "all signals entering or leaving the sequence of HDL code delineated by the begin block and end block statements." A1144(14:20-25). This allows a designer "to select many parts of the HDL source [code] for probing without typing a text statement probe command for each line of HDL." A1144(14:37-40). *Compare* A1116-18 Figs. 12 & 14 (non-instrumented code and resulting circuit schematic) *with* A1120-22 Figs. 16 & 18 (instrumented code and resulting circuit schematic).

typical gate-level simulation, it is "difficult, if not impossible" to "map[]" output values (0 or 1) of corresponding circuit components "to particular [HDL] source code lines." A119(2:7-9). To address this problem, the '376 patent attempts to make it easier for a designer to preserve information in the HDL source code that otherwise would be lost during synthesis, and to determine the location of the source code that is necessary for debugging.

Specifically, the patent teaches a method of adding instrumentation that identifies lines of source code and their components in the circuit for later tracing, and collects information through simulation that otherwise would be lost. A121-25(5:27-30, 5:38-40, 7:24-27, 8:60-64, 13:60-67). Through instrumentation, additional data are produced that "identif[y] the RTL source code statements each instrumentation output signal is associated with." These are called the "[i]nstrumentation data." A121(6:32-34).

At issue here is the patentability of claims 1 and 28 of the '376 patent. Claim 1 recites:

> A method comprising the steps of
>
> a) identifying at least one statement within a register transfer level (RTL) synthesizable source code; and
>
> b) synthesizing the source code into a gate-level netlist including at least one instrumentation signal, wherein the instrumentation signal is indicative of an execution status of the at least one statement.

A126(15:2-9). Similarly, claim 28 recites:

> A storage medium having stored therein processor executable instructions for generating a gate-level design from a register transfer level (RTL) synthesizable source code, wherein when executed the instructions enable the processor to synthesize the source code into a gate-level netlist including at least one instrumentation signal, wherein the instrumentation signal is indicative of an execution status of at least one synthesizable statement of the source code.

A127(17:61-18:7).

III. PROCEDURAL BACKGROUND

This proceeding is one of several involving Synopsys and Mentor, which compete in the field of verifying circuit designs. In this dynamic field, the marketplace has changed rapidly in recent years. Litigation also has been frequent, including patent litigation between Mentor and

Synopsys that has been ongoing since 2010. *Supra* at xiii-xiv. Among the patents being litigated is the '376 patent, on which Synopsys sought a declaratory judgment of non-infringement, and Mentor sued for infringement. *Id.* In 2012, prior to Synopsys' declaratory judgment action, Synopsys petitioned for *inter partes* review of the '376 patent. This appeal arises from that proceeding.

A. The Board Institutes *Inter Partes* Review Of The '376 Patent.

In 2011, Congress enacted the America Invents Act (AIA), Pub. L. No. 112-29, 125 Stat. 284 (2011), which created new methods for the Board (also newly created) to hear adversarial challenges to the patentability of issued patents. Relevant here is the *inter partes* review, which authorizes "a person who is not the owner of a patent" to ask the Board "to cancel as unpatentable 1 or more claims of a patent … on a ground that could be raised under [35 U.S.C. §] 102 or 103." 35 U.S.C. § 311(a)-(b).

Inter partes review is instituted when the PTO determines "that there is a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the claims challenged in the petition." *Id.* § 314(a). Upon instituting review, the Board considers the patentability

of the challenged patent. *See generally id.* § 316; 37 C.F.R. pt. 42. The Board holds what is called a "trial" (in actuality, more akin to oral argument), after which "the Board shall issue a final written decision with respect to the patentability of any patent claim challenged by the petitioner." 35 U.S.C. § 318(a).

Synopsys filed a petition for *inter partes* review that challenged claims 1-15 and 20-33, contending that they were anticipated and obvious in view of five prior-art references, including Gregory. A145-51. The Board instituted review of claims 1-9, 11, and 28-29, finding a reasonable likelihood that they were anticipated by Gregory. A23-28.

As a threshold matter, the Board rejected Mentor's counterargument that review was precluded by 35 U.S.C. § 315(b). A15-18. Section 315(b) provides that "review may not be instituted if the petition requesting the proceeding is filed more than 1 year after the date on which the petitioner, real party in interest, or privy of the petitioner is served with a complaint alleging infringement of the patent." Mentor argued that this bar applies because Mentor sued EVE on the '376 patent years earlier, and Synopsys acquired EVE shortly after filing its petition for *inter partes* review. But, the Board explained, "the plain

language of § 315(b)" considers privity at the time the district court complaint is served, and there was no evidence that Synopsys and EVE were in privity at that time, or indeed even when the petition was filed. A15-16.

Thus, the Board went on to determine that there was "a reasonable likelihood" that "at least 1 of the claims challenged in the petition" was invalid, 35 U.S.C. § 314(a)—namely, claims 1-9, 11, and 28-29. A23-28. The Board, however, "decline[d] to institute *inter partes* review on claims 10, 12-15, 20-27, and 30-33." A14, 18-40.

B. The Board Cancels Claims 5, 8, And 9.

At the conclusion of its review, the Board issued a final written decision in which it canceled claims 5, 8, and 9, but not claims 1-4, 6-7, 11, and 28-29. The Board reiterated its rejection of Mentor's argument that the "proceeding is barred" by § 315 and should not have been instituted. A52-55. It also rejected Mentor's contentions that Synopsys is not the real party-in-interest, and that assignor estoppel precludes *inter partes* review. A55-58.

The Board next addressed claim construction. The claims at issue in this appeal all use the phrase "instrumentation signal."

A126-27(15:2-8, 17:60-18:7) ('376 patent). As the Board explained, Mentor and Synopsys generally agreed that "instrumentation signal" means "an output … created during synthesis of RTL source code by inserting additional code into a program that indicates whether the corresponding RTL source code statement is active." A67-68. The parties disagreed, however, about whether the "output" can only result from an *increased number* of circuit components (as Mentor argued), or if instead (as Synopsys contended) the output also could result from *preserving already-existing* circuit components. A67.

The Board agreed with Synopsys that under "the broadest reasonable construction of 'instrumentation signal,'" the term can be satisfied either by increasing the number of circuit components *or* by "preserving already existing circuit components." A67. In rejecting Mentor's construction, the Board noted that the '376 patent discloses two primary embodiments—one in which "instrumentation results in generating a modified gate-level design," and another in which the "source code is analyzed … without modifying the gate-level design," A121(5:17-44)—and that Mentor's proposed construction is inconsistent with the latter embodiment, A63-65.

As to "execution status," the Board "determine[d] that the broadest reasonable construction of 'execution status' in light of the '376 patent specification is information regarding whether a particular HDL instruction has been performed." A68.

Turning to invalidity, the Board concluded that claims 5, 8, and 9 are anticipated by Gregory. A74-79, 81-82. This is because Gregory, at a minimum, preserves circuit components during the optimization process. "Gregory discloses all the limitations of claim 5 arranged as recited in the claim" by "disclos[ing]" a method where the "synthesis process[] injects information into the resulting netlist when a probe is encountered, which results in components of the final circuit that are traceably related to the source code." A79; *accord* A81-82. The Board also denied Mentor's contingent motion to amend claims 5, 8, and 9 because Mentor did not satisfy its burden of demonstrating the validity of its proposed amended claims. A88-90.

As to claims 1 and 28, however, the Board found them not to be anticipated by Gregory. Mentor argued that Gregory does not anticipate because it does not teach adding circuit components, and the Board rejected that argument as based on the unduly narrow claim construc-

tion it had rejected. A71. But, the Board concluded that claim 1 is not anticipated because it believed Gregory does not disclose a method of instrumenting code that indicates the "execution status" of at least one of the instrumented source code statements. A72-73. The Board rejected Synopsys' explanations of how Gregory indicates the "execution status" of the instrumented code because, the Board said, those explanations were rooted in disclosures in Gregory itself rather than "expert testimony" or other "evidence." A72, 74. And, because claim 28 contains the same "execution status" limitation, the Board held that Gregory does not anticipate it either. A74.

SUMMARY OF ARGUMENT

I. Gregory anticipates claims 1 and 28 of the '376 patent. Only two claim limitations are at issue. The Board correctly held that Gregory teaches "instrumentation signal," but erred in concluding that Gregory does not disclose "execution status."

A. An "instrumentation signal" is produced either by adding components to the circuit or by "preserving already existing circuit components." A67-68. Mentor did not dispute that Gregory discloses a method that preserves circuit components so they are not removed

during optimization. Nor could it. Gregory graphically illustrates this through figures in which circuit components associated with instrumented source code are not optimized away, whereas those same components disappear when non-instrumented but otherwise identical source code is optimized. *Infra* at 39.

Moreover, even under Mentor's erroneous construction (that an "instrumentation signal" can be created only by adding additional circuit components), Gregory teaches this limitation by "provid[ing] a method" that "artificially inject[s] primary inputs or outputs into the initial circuit," A1141(8:11-17), and "adds additional information or components to the initial circuit," A69-70 (citing A1141(8:26-30) (Gregory '109)). This is evident from Gregory, which shows that additional circuit components are produced during the synthesis of instrumented code. *Infra* at 42-46.

B. The Board erred in concluding that Gregory does not disclose a method indicating whether a line of source code "has been performed"—i.e., "execution status." A68. Specifically, the pair of schematics in Gregory Figures 16 and 18 discloses execution status—they show that, by reviewing the temporary outputs that result from instru-

menting the circuit's source code, one can determine which portion of a logical statement was executed. Figures 8 and 9 show the same thing—and indeed, disclose execution status two different ways. In each of these sets of figures, inputs and outputs can be traced through the circuit schematic to determine which statement was executed. *Infra* at 54-61.

The Court need not undertake this analysis, however, because the Board's failure to find Gregory anticipatory was based on fundamental, underlying legal errors that themselves require vacatur and remand. The Board incorrectly adopted a requirement that the prior art must "explicitly" disclose a claim limitation, despite this Court's "well settled" law that "express" disclosures are not necessary to anticipate. *In re Cruciferous Sprout Litig.*, 301 F.3d 1343, 1349 (Fed. Cir. 2002); *infra* at 47-50. The Board also declined to consider whether Gregory's teachings disclose "execution status" because Synopsys did not present expert testimony along with the Gregory disclosure. But "there is no invariable requirement that a prior art reference [must] be accompanied by expert testimony" to anticipate. *Meyer Intellect'l Props. Ltd. v. Bodum, Inc.*, 690 F.3d 1354, 1374 (Fed. Cir. 2012) (quotation marks

and brackets omitted); *infra* at 62-63. And, the Board declined to consider portions of Synopsys' argument based on an erroneous waiver theory that these arguments were raised in Synopsys' reply brief, even though that was the first opportunity to respond to arguments raised by Mentor. *Infra* at 65-67.

For each of these reasons, the Board's determination that claims 1 and 28 of the '376 patent are not anticipated by Gregory and must be reversed or vacated.

II. The Board erred as a matter of law in issuing a final written decision that failed to address "the patentability of any patent claim challenged by" Synopsys. 35 U.S.C. § 318(a). The Supreme Court and this Court repeatedly have held that "any" means "all." *E.g.*, *United States v. Rosenwasser*, 323 U.S. 360, 363 (1945). The final written decision, however, did not address all challenged claims. This basic error requires that this aspect of the final written decision be vacated and remanded with instructions to the Board to issue a final written decision that complies with the clear statutory requirement.

STANDARD OF REVIEW

This Court reviews the Board's "legal conclusions *de novo*, and [its] factual findings underlying those determinations for substantial evidence." *K/S HIMPP v. Hear-Wear Techs., LLC*, 751 F.3d 1362, 1364 (Fed. Cir. 2014). Anticipation "is a question of fact … review[ed] for substantial evidence." *ClearValue, Inc. v. Pearl River Polymers, Inc.*, 668 F.3d 1340, 1343 (Fed. Cir. 2012). "Substantial evidence is more than a scintilla." *Nippon Steel Corp. v. United States,* 458 F.3d 1345, 1351 (Fed. Cir. 2006) (alteration and quotation marks omitted). When reviewing for substantial evidence, the "court must consider the record as a whole, including that which fairly detracts from its weight, to determine whether there exists such relevant evidence as a reasonable mind might accept as adequate to support a conclusion." *Id.* (quotation marks omitted); *Institut Pasteur & Universite Pierre et Marie Curie v. Focarino*, 738 F.3d 1337 (Fed. Cir. 2013) (vacating PTO decision for lack of substantial evidence); *In re Rambus*, *Inc.*, 753 F.3d 1253, 1255-57 (Fed. Cir. 2014) (same).

ARGUMENT

I. GREGORY ANTICIPATES CLAIMS 1 AND 28.

"In an inter partes review," a petitioner must prove "unpatentability by a preponderance of the evidence." 35 U.S.C. § 316(e). At issue here is anticipation. A patent is anticipated if "the invention was described" in another "patent granted on an application … filed … before the invention by the applicant." 35 U.S.C. § 102(e) (2010).[9] That condition is satisfied when an earlier patent "discloses all of the claim limitations" of the patent-at-issue. *Am. Calcar, Inc. v. Am. Honda Motor Co.*, 651 F.3d 1318, 1342 (Fed Cir. 2011).

Specifically at issue are claims 1 and 28 of Mentor's '376 patent. Because Mentor's responses regarding anticipation were the same for both claims, *see, e.g.*, A277; A463, and because the Board treated the claims the same, A74, this brief treats claim 1 as representative. It recites:

[9] Because the '376 patent was issued before the AIA's amendments to § 102, its "earlier version … govern[s]." *Solvay S.A. v. Honeywell Int'l Inc.*, 742 F.3d 998, 1000 n.1 (Fed. Cir. 2014).

A method comprising the steps of

a) identifying at least one statement within a register transfer level (RTL) synthesizable source code; and

b) synthesizing the source code into a gate-level netlist including *at least one instrumentation signal*, wherein the instrumentation signal is indicative of an *execution status* of the at least one statement.

A126(15:2-8) (emphasis added to the limitations-at-issue). It is undisputed that Gregory discloses the elements "identifying at least one statement within a register transfer level (RTL) synthesizable source code," and "synthesizing the source code into a gate-level netlist." Thus, before the Board were two limitations—whether Gregory discloses "including at least one instrumentation signal," and whether that instrumentation signal "is indicative of an execution status." The Board correctly determined that Gregory discloses an "instrumentation signal," *infra* Section I.A., but erred in concluding that that instrumentation signal is not "indicative of an execution status," *infra* Section I.B.

A. Gregory Discloses "Instrumentation Signal" Under Every Proposed Claim Construction.

1. The Board correctly found that Gregory discloses an "instrumentation signal." A71. We begin with this claim term—notwithstanding that the Board's determination was correct—because

understanding this term is important to understanding the invention and because "instrumentation signal" relates closely to "execution status," about which the Board erred. *See* A126(15:2-8) ('376 patent) (claim 1: "the *instrumentation signal* is indicative of an *execution status* of the at least one statement" (emphases added)).

In *inter partes* review, the Board interprets "claim terms … according to their broadest reasonable construction." A58 (final written decision) (citing 37 C.F.R. § 42.100(b); Office Patent Trial Prac. Guide, 77 Fed. Reg. 48,756, 48,766 (Aug. 14, 2012)). Here, the Board determined that the "broadest reasonable construction" of this term "at least includes an output signal created during synthesis of RTL source code by inserting additional code into a program that indicates whether the corresponding RTL source code statement is active." A67-68. That is to say, an "instrumentation signal" is a signal or output that (a) indicates which line (or lines) of the "if-then-else" statements have been performed, and (b) is produced when additional commands are added to the circuit's source code specification before synthesis. The Board explained that such an output could be achieved in two ways: by adding additional

components to the circuitry, or "by preserving already existing circuit components." A67.

Under this construction, the Board held that Gregory discloses an "instrumentation signal," and Mentor did not argue otherwise. A71-72 (final written decision) (rejecting Mentor's "instrumentation signal" arguments as based on its proposed and rejected construction; citing A444-49 (Mentor's Response Brief)). This was with good reason: Gregory plainly discloses a method by which existing circuit components are preserved. Specifically, Gregory discloses instrumenting source code in a way that tells the synthesizer and the optimizer not to remove or replace circuit components associated with the instrumented code. In Gregory's own words, Gregory teaches a method of "mark[ing] the synthesis source code … such as [with a] 'probe', along with some additional optional information," A1141(8:21-26), so that, during the process of synthesis, "[w]hen the translator[10] encounters a probe

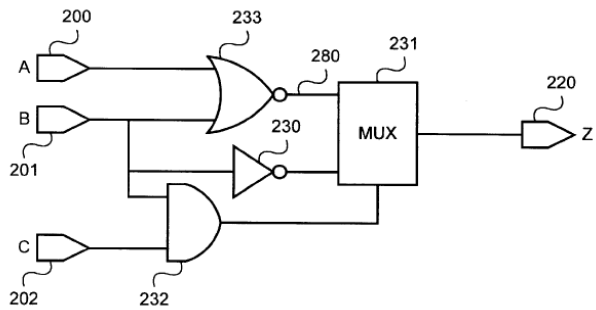
[10] A translator is another name for the synthesizer or synthesizing software, which "converts" the "HDL[] description of a desired [circuit] function … into gates and other circuit structures that directly correspond statement by statement with the designer's description." A1138(2:59-3:3) (Gregory '109).

statement, the translator interjects information … to indicate to the optimizer that optimization should not proceed 'past' or 'through' that particular point." A1143(11:20-23). The effect is that when the circuit is optimized (i.e., when unnecessary circuit components are removed), the information marked by the probe will be preserved. A69-70 (final written decision).

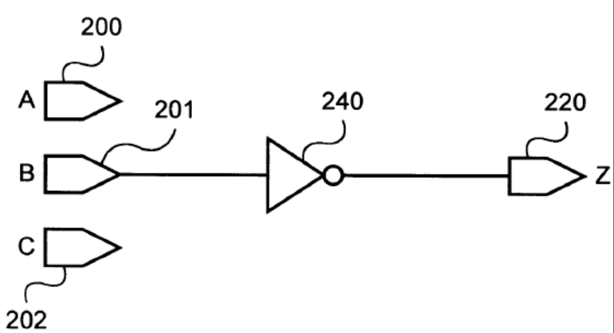
That this process causes existing circuit components to be preserved is demonstrated graphically in Gregory Figures 6 and 7, and 9 and 10. All four figures depict circuit schematics resulting from synthesizing and optimizing the same underlying source code—or, more precisely, from what started as the same underlying source code, shown in Figure 4. A1142(9:48-59) (Gregory '109). One pair of diagrams shows what happens when that source code is synthesized (Figure 6) and optimized (Figure 7) *without* having been instrumented. A1142(9:49-54), A1143(12:1-42). The second set of diagrams shows what happens when the same underlying source code is instrumented through the addition of a probe statement (Figure 8), then synthesized (Figure 9), and optimized (Figure 10). A1143-44(12:43-13:15). The addition of the probe causes circuit components that were optimized

away in Figure 7 to be preserved in Figure 10 (as reflected in the highlighting below):

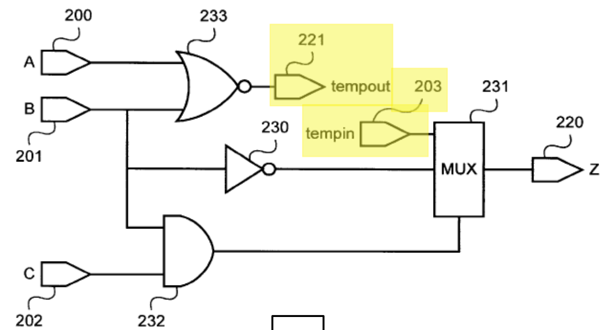
(Fig. 6: synthesized circuit *without* instrumentation)



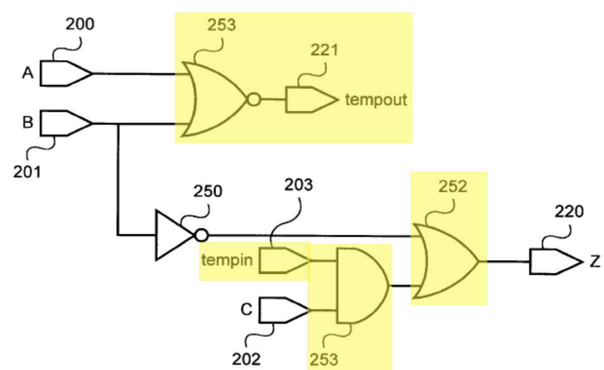
(Fig. 7: optimized circuit *without* instrumentation)



(Fig. 9: synthesized circuit *with* instrumentation)



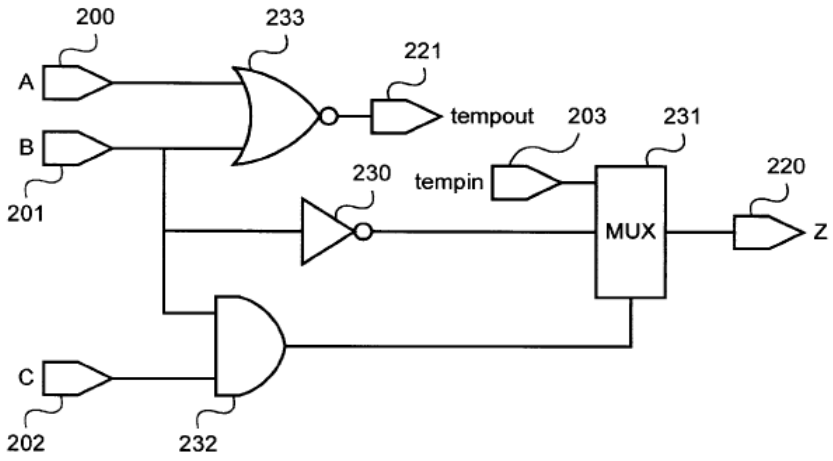
(Fig. 10: optimized circuit *with* instrumentation)



A1110-14 (instrumentation signals and preserved components highlighted); *see* A1142(9:48-60) (Gregory '109); *compare* A1108, 1112 Figs. 4, 8 (non-instrumented and instrumented source code); *see also*

A69-70 (final written decision) (explaining that Gregory teaches a method that prevents "'certain [circuit] components'" associated with the instrumented code from "'be[ing] replaced during optimization'" (quoting A1141(8:26-30) (Gregory))).

One example of an "instrumentation signal" is the "tempout" in Figure 9. A71, 79 (final written decision) ("'tempout' [in Gregory Figure 9] qualifies as an 'instrumentation signal[]'"). Gregory explains that "tempout" or "temporary output," A1144(13:1-3), is the signal produced by a logical operation in the circuit as a result of instrumenting the code, A1141(8:11-17). The output is "temporary" not because it vanishes, but because it is not the final output ("Z") of the circuit. Accordingly, Figure 9, which depicts a circuit schematic that results when certain *instrumented* source code (set forth in Figure 8) is synthesized, contains a "tempout":



A1113 Fig. 9. By comparison, Figure 6, which is the synthesized circuit of non-instrumented but otherwise identical source code, contains no such temporary output. *See* A1110. And, as illustrated in the comparison above—between Gregory Figures 9 and 10 (which contain "tempout"), and Figures 6 and 7 (which do not)—adding the "tempout" signal to the circuit preserves information and logic gates that would otherwise be optimized away. "Tempout" is therefore an instrumentation signal, the Board explained, because it preserves information or logic in the circuitry, and is produced as a result of instrumenting the underlying source code with a probe statement. A71, 79 (final written decision); *see also* A1142(9:48-57) (Gregory '109) (describing these figures).

The Board's conclusion that "tempout" in Gregory Figure 9 is an "instrumentation signal" is confirmed by Mentor's own patent. In the '376 patent, Figure 10 adds four outputs, Sig_Trace1 through Sig_Trace4. A107. Sig_Trace1 and Sig_Trace2 are similar to "tempout" in Gregory in that they preserve logic gates and carry outputs. They do so, as the Board explained, without adding any "additional logic gates," A63—and Mentor's own patent itself calls them "instrumentation

signals," A123(9:12-14). Thus, as described by the '376 patent itself, signals generated during synthesis as a result of instrumenting the circuit's source code are "instrumentation signals," even if those signals did not result from adding components to the circuitry. The Board was correct that "tempout" in Gregory is an "instrumentation signal." A63-64.

2. Mentor does not dispute that Gregory discloses a method of instrumenting code to generate a signal by preserving circuit components. *See* A71 (final written decision) (describing Mentor's argument). Instead, Mentor has argued that Gregory is not anticipatory because it does not disclose a method for adding "additional logic [gates]" to the circuitry. *Id.* The Board rejected this argument as premised on a "narrower construction of the term 'instrumentation signal' than we have adopted," because it assumes that an "instrumentation signal" can only be created by adding—not by preserving—circuit components. A67-68, 71.

The Board rightly rejected Mentor's claim construction because it is inconsistent with the specification of the '376 patent, not least because it reads out one of that patent's primary embodiments. A59-68;

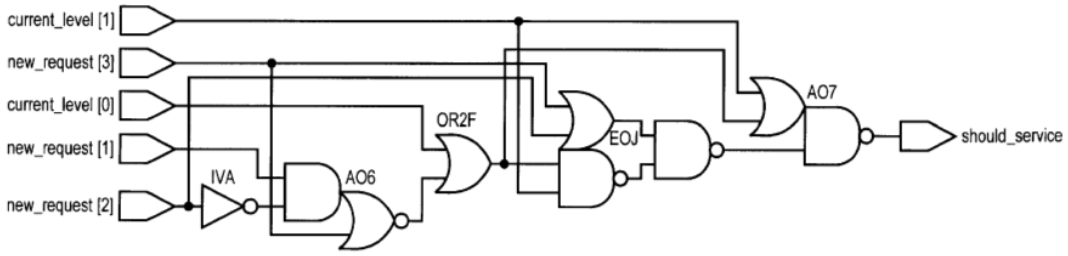
supra at 27. But even if Mentor's claim construction were correct—such that an "instrumentation signal" required adding logic to the circuit—Gregory still meets this limitation because it discloses a method for generating instrumentation signals by adding circuit components. As Gregory explains: Its "invention provides a method for introducing additional points in the [circuit] design … by artificially injecting primary inputs or outputs into the initial circuit." A1141(8:11-17). By instrumenting the code, the synthesizer and optimizer produce what Gregory describes as "a new circuit subject to the[] constraints" that were added when the code was instrumented. A1143(11:24-26); *accord* A1144(14:55-57).

For example, "tempout" in Gregory Figures 9 and 10 is a signal that is produced by instrumenting Gregory's source code—as demonstrated by the fact that this "tempout" signal does not exist in the circuits in Figures 6 and 7, which are produced from *non-instrumented* code. A1110-14 Figs. 6-7, 9-10. Adding this probe statement, the Board found, "adds additional information or components to the initial circuit." A69-70 (citation omitted). Thus, the Board explained, Gregory describes a method

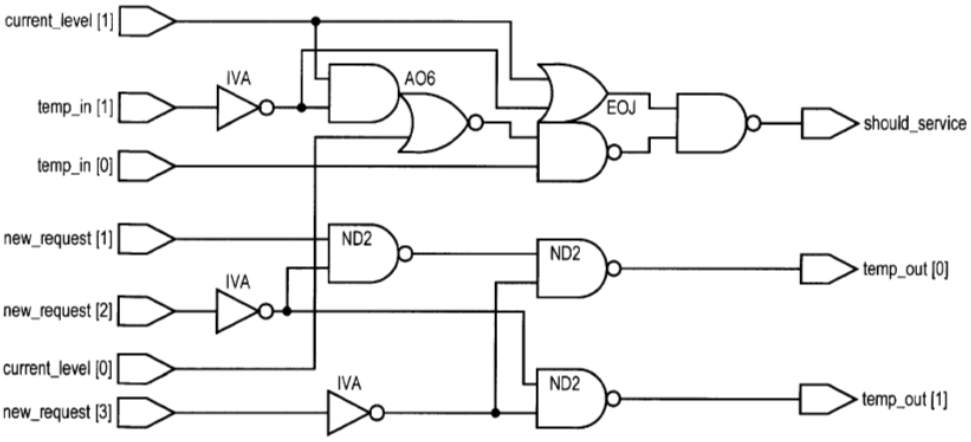
> that "interjects information into the netlist" when [the synthesizer] encounters a probe statement and "generates a circuit that provides the same function as it did without the 'probe' statement, *but adds additional information or components to the initial circuit* that indicate that certain components should not be replaced during optimization." These components "are traceably related to the source HDL" and they "facilitate debugging."

Id. (emphasis added; brackets and citation omitted) (quoting A1141, 1143(8:26-41, 11:20-24) (Gregory '109)).

That this element is disclosed even under Mentor's erroneous claim construction is further evident from the example in Gregory Figures 14-18. *See* A79 (Board finding that "temp_out" in Gregory Figure 18, like "tempout" in Gregory Figure 9, adds "information" to the circuit design). As discussed above (at 19-21 & n.8), the source code synthesized into Gregory Figures 9 and 10 is instrumented by adding a single probe statement that identifies a single line of code. The same is evident by comparing Gregory Figures 14 and 18. Figure 14 depicts a schematic of a circuit that has been optimized from certain non-instrumented source code, A1142(9:64-10:2):



A1118 Fig. 14. The circuit in Figure 18 is optimized from source code that would be the same, A1142(9:64-10:11), except that it was instrumented:



A1122 Fig. 18. Because the code was instrumented, Figure 18 has two more logic gates after optimization than Figure 14. *See* A812-13 (Synopsys trial presentation; discussing A781 (demonstrative exhibit)); *compare* A1118, 1122 Figs. 14, 18 (Gregory '109); *see also* A1410-14 ¶¶ 25, 31-36 (Synopsys' Expert's Decl.) (explaining in the context of Mentor's motion to amend that Figure 18 has two more gates than Figure 14). Thus, Figure 18 depicts two instrumentation signals

("temp_out[0]" and "temp_out[1]"), which, along with the added circuit components, are generated as a result of instrumenting the underlying source code.

Under any construction of "instrumentation signal," Gregory discloses this element.

B. Gregory Discloses "Execution Status."

The limitation of the '376 patent erroneously resolved by the Board was "execution status." Recall that the '376 patent claims a method of identifying at least one "statement" in the source code, then "synthesizing the source code into a gate-level netlist including at least one instrumentation signal, *wherein the instrumentation signal is indicative of an execution status*" of the identified statement. A126(15:2-8) ('376 patent) (emphasis added). We have just explained why the Board was correct to find that Gregory teaches adding at least one "instrumentation signal." The question, then, is whether Gregory's instrumentation signals are "indicative of an execution status" of the identified statement. *Id.* The answer is clearly yes.

The Board construed "execution status" to mean "information regarding whether a particular HDL instruction has been performed."

A68. In other words, "execution status" is information about whether the identified "statement" (i.e., an instruction line) of source code has been performed in the circuit. The Board determined that Gregory does not disclose this limitation. In so doing, it committed a series of legal errors that require vacatur at the threshold, in addition to erring about what Gregory discloses.

1. The Board erred in requiring an "explicit" disclosure.

The Board erred as a matter of law in its analysis, and that legal error requires vacatur and remand. According to the Board, Gregory does not disclose "execution status" because it "does not *state explicitly*" that any particular element "indicates an 'execution status.'" A73 (emphasis added). But Gregory did not need to utter the magic words "execution status" to be anticipatory. This Court has made clear that a prior-art "reference need not satisfy an *ipsissimis verbis* test," *In re Gleave*, 560 F.3d 1331, 1334 (Fed. Cir. 2009)—i.e., requiring "the same terminology [to be] in the prior art in order to find anticipation," *Akzo N.V. v. ITC*, 808 F.2d 1471, 1479 n.11 (Fed. Cir. 1986); *see also In re Bond*, 910 F.2d 831, 832 (Fed. Cir. 1990); *Structural Rubber Prods. Co. v. Park Rubber Co.*, 749 F.2d 707, 716 (Fed. Cir. 1984).

On the contrary, it is "well settled that a prior art reference may anticipate when the claim limitations [are] not expressly found in that [prior-art] reference." *Cruciferous Sprout Litig.*, 301 F.3d at 1349. For instance, "[a]nticipation can occur when a claimed limitation is 'inherent' or otherwise implicit in the relevant [prior-art] reference." *Standard Havens Prods., Inc. v. Gencor Indus., Inc.*, 953 F.2d 1360, 1369 (Fed. Cir. 1991); *accord Glaverbel Societe Anonyme v. Northlake Mktg. & Supply, Inc.*, 45 F.3d 1550, 1554 (Fed. Cir. 1995); PTO, *Manual of Patent Examining Procs.* § 2112, *available at* http://tinyurl.com/pdjq8uy ("The express, implicit, and inherent disclosures of a prior art reference may be relied upon in the rejection of claims under 35 U.S.C. § 102 ….").[11]

---

[11] *See also, e.g.*, *King Pharm., Inc. v. Eon Labs, Inc.*, 616 F.3d 1267, 1274 (Fed. Cir. 2010) (affirming invalidity based on anticipation due to inherent components of the prior art); *Schering Corp. v. Geneva Pharm., Inc.*, 339 F.3d 1373, 1377-80 (Fed. Cir. 2003) (finding inherent anticipation even though "the prior art supplie[d] no express description of any part of the claimed subject matter"); *In re Montgomery*, 677 F.3d 1375, 1379-83 (Fed. Cir. 2012) (affirming that a patent was anticipated based on prior art's inherent teaching). This Court has repeatedly overturned jury verdicts based on not-explicit teachings in prior art. *Exergen Corp. v. Wal-Mart Stores, Inc.*, 575 F.3d 1312, 1318-20 (Fed. Cir. 2009); *Ecolab, Inc. v. FMC Corp.*, 569 F.3d 1335, 1345-48 (Fed. Cir. 2009).

Similarly, and of particular relevance here, this Court has held for more than 80 years that illustrations can be anticipatory teachings even when the written description is otherwise silent. *See In re Bager*, 47 F.2d 951, 953 (CCPA 1931) ("Description for the purposes of anticipation can be by [prior art] drawings alone as well as by words" (internal quotation marks omitted)); *In re Wagner*, 63 F.2d 987, 988 (CCPA 1933) (holding the figures in the prior art "sufficient" to demonstrate anticipation); *accord In re Mraz*, 455 F.2d 1069, 1072 (CCPA 1972); *Therasense, Inc. v. Becton, Dickinson & Co.*, 593 F.3d 1325, 1334-36 (Fed. Cir. 2010) (finding obviousness based on figures in the prior art).

The rationale for this rule is simple, and derives from a core requirement for patentability: The novelty requirement of § 102 means that a patent can only be issued for a new invention. *See Hoover Group, Inc. v. Custom Metalcraft, Inc.*, 66 F.3d 299, 302 (Fed. Cir. 1995); 1-3 *Chisum on Patents* § 3.01 (2014). Regardless of the terminology used—and whether the disclosure is "express," "explicit," "inherent," or "implicit"—so long as a claimed invention has already been disclosed, its "subject matter [is] in the public domain" and is not "new." *Schering*

Corp., 339 F.3d at 1379; *see also Standard Havens*, 953 F.2d at 1369; *Gleave*, 560 F.3d at 1334; *Bager*, 47 F.2d at 953.

By departing from this Court's settled authority, the Board applied the wrong legal standard, and that error requires vacatur and remand for reconsideration under a proper legal standard. "'[A] simple but fundamental rule of administrative law is that a reviewing court, in dealing with a determination or judgment which an administrative agency alone is authorized to make, must judge the propriety of such action solely by the grounds invoked by the agency,'" because "'[i]f those grounds are inadequate or improper, the court is powerless to affirm the administrative actions.'" *Burlington Truck Lines, Inc. v. United States*, 371 U.S. 156, 169 (1962) (alterations omitted) (quoting *SEC v. Chenery Corp.*, 332 U.S. 194, 196 (1947)). That basic principle of course applies equally to the PTO. *Gechter v. Davidson*, 116 F.3d 1454, 1457-60 (Fed. Cir. 1997). The Board's decision therefore must be vacated.

2. Gregory discloses "execution status."

Not only did the Board use an erroneous legal standard; it also erred because Gregory indeed discloses a method that "indicates an 'execution status' of an HDL instruction." A73. Gregory does so in at

least two different ways. Two sets of figures (8-10 and 16-18) separately illustrate how Gregory's method of instrumenting source code indicates the "execution status" of the identified source code statement. The Board failed to address one of these disclosures and erred in its treatment of the other.

a. Figures 16 and 18 of Gregory disclose "execution status."

Gregory teaches how to instrument a line of code using a probe statement and a block of code. *Supra* at 21 n.8. Using a block probe, a circuit designer can mark "all" of the lines of code within an identified "sequence of HDL code." A1144(14:20-25); *see* A1120 Fig. 16. Just as in the '376 patent, Gregory's invention tells "[t]he translator [to] create temporary inputs … and temporary outputs," thereby preserving the logical operations described in the instrumented source code. A1144(14:26-29). The designer then can examine the temporary outputs (i.e., look at the signals that flow from the circuit) and, based on that information, know which portion of a logical statement ("then" or "else") was executed. Thus, the invention provides "information regarding whether a particular HDL instruction has been performed." *See* A68 (final written decision).

i. We explain below (at 54-58) how Figures 16 through 18 demonstrate "execution status," but the Court need not even reach that issue. The Board's decision must be vacated for the simple reason that, when the Board assessed execution status in its final written decision, it failed to address Synopsys' argument. Synopsys raised Figures 16 through 18 in its petition for *inter partes* review regarding claim 28 to demonstrate Gregory's anticipatory teaching of "execution status," A180-81, and as discussed above (at 34-35), the parties and the Board treated claims 1 and 28 in tandem throughout the proceedings below. The Board did consider Figures 16 and 18 when it invalidated claims 5, 8, and 9 of the '376 patent, yet it failed to do so for claim 1, a form of inconsistent treatment that gives rise to a risk of inconsistent decisions. A75-79, 81-82.

The Board's failure to address this argument was error. *See generally Q.I. Press Controls, B.V. v. Lee*, 752 F.3d 1371, 1383-84 (Fed. Cir. 2014) (vacating the Board's decision of non-obviousness because the Board failed to consider a prior-art reference cited as to certain claims when that reference was relevant to other claims containing only "minor differences"). Consistent with the fundamental administrative-

law principle discussed above (at 50, 52-53), when, as here, an agency fails to consider a party's argument, the reviewing court can only vacate and remand. *Burlington Truck Lines*, 371 U.S. at 168-69 ("an agency's discretionary order [may] be upheld, if at all, [only] on the same basis articulated in the order by the agency itself"); *Motor Vehicle Mfrs. Ass'n, Inc. v. State Farm Mutual Auto. Ins. Co.*, 463 U.S. 29, 48-50 (1983) (a "sufficient" basis to reject an agency's decision is that it "submitted no reason[] at all" in response to a party's argument); *Gechter*, 116 F.3d at 1457-60 (collecting cases; vacating and remanding a PTO decision because an agency decision "must contain sufficient findings and reasoning to permit meaningful appellate scrutiny"); *In re Lee*, 277 F.3d 1338, 1342 (Fed. Cir. 2002) (an agency "must set forth its findings and the grounds thereof, as supported by the agency record, and explain its application of the law to the found facts").[12]

---

[12] *Accord Iowa v. FCC*, 218 F.3d 756, 759 (D.C. Cir. 2000) ("the Commission's failure to address Iowa's argument requires that we remand this matter for the Commission's further consideration"); *TNA Merchant Projects, Inc. v. FERC*, 616 F.3d 588, 592-93 (D.C. Cir. 2010); *Darrell Andrews Trucking, Inc. v. Fed. Motor Carrier Safety Admin.*, 296 F.3d 1120, 1134-35 (D.C. Cir. 2002).

ii. Gregory indeed discloses a method of instrumenting code to create instrumentation signals that indicate "execution status"—i.e., whether particular HDL instructions have been executed. This is clear from Gregory Figure 16, which discloses the following instrumented source code:

```
--Synopsys block_probe_begin
decode: process(new_request)
begin
 if(new_request(3) = '1') then
  new_level <= "11";
 elsif(new_request(2) = '1') then
  new_level <= "10";
 elsif(new_request(1) = '1') then
  new_level <= "01";
 else
  new_level <= "00";
 end if;
end process;
--Synopsys block_probe_end
```

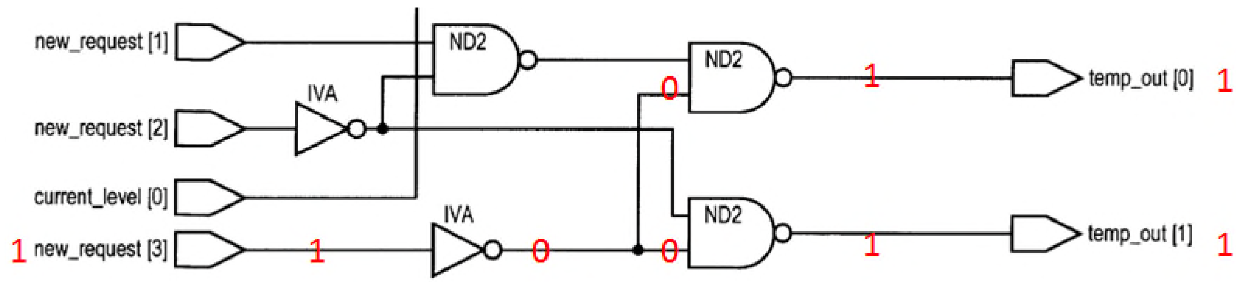
A1120 Fig. 16. This code is much like the if-then-else condition set forth elsewhere in Gregory and described above (at 18-20), just with additional levels of instructions. At each step, "if" the input equals "1," "then" certain temporary outputs will result and the process ends; otherwise, the operation proceeds to the next line. Those temporary outputs are represented by a two-digit number in double quotation

marks. For each two-digit number, the first digit refers to temp_out[1], and the second refers to temp_out[0], in Figure 18.

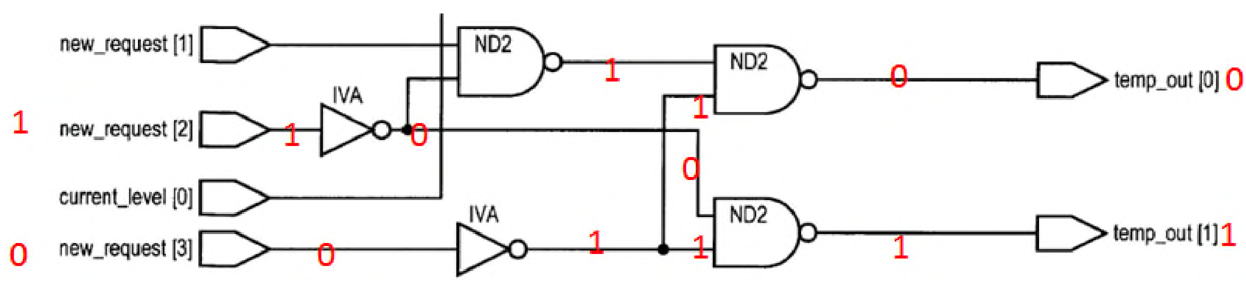
Each of the four inputs is associated with a *unique* temporary output: 11, 10, 01, or 00. This means that by looking at the output signals, one can know with certainty which statement was executed. To verify that this is so, each input can be traced through the logic of the circuit (using the basic logical analysis described above, *supra* at 11-12); and conversely, the temporary outputs can be used to backtrack through the schematic to determine which statement in the code was executed—i.e., to identify execution status. Gregory therefore discloses execution status.

The circuit schematic confirms it. So, for instance, the first line of code in Figure 16 directs that if the input titled new_request[3] equals 1, then temp_out[1] and temp_out[0] both will equal 1. Tracing the inputs through the Figure 18 schematic confirms this, as demonstrated below through the red numerals that we have added to show the logical operations of each gate. The logic gates ND2 are NAND gates: If either input is a 0, then the output is a 1. *Supra* at 11. Similarly, retracing the outputs back through the circuit (i.e., going from right to left), if the

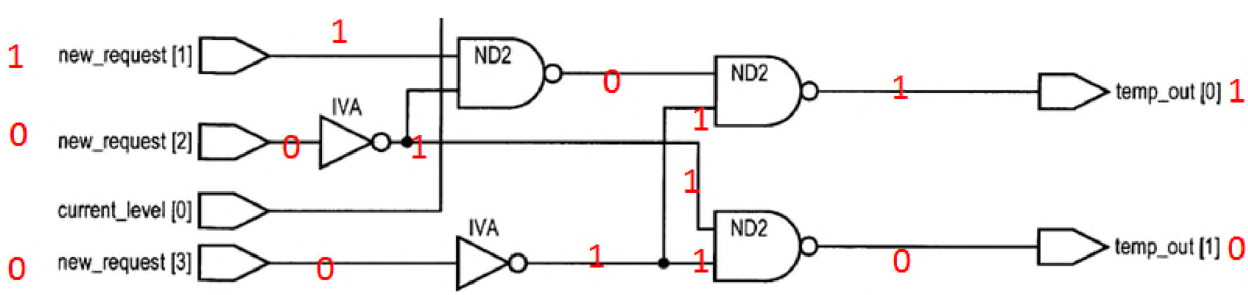
outputs of ND2 are 11, then there must have been an input of 1 for new_request[3]:



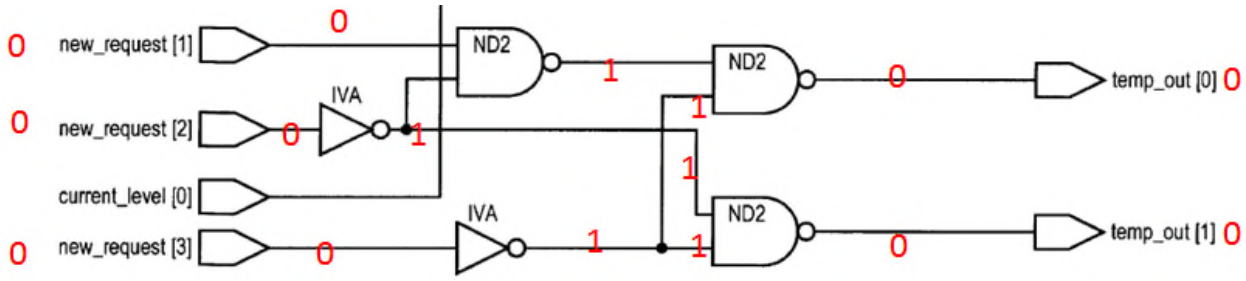
Next, according to the code in Figure 16, if new_request[3] does *not* equal 1 (“else”), but new_request[2] does equal 1, then temp_out[1] equals 1 and temp_out[0] equals 0. Again, the circuit bears this out:



If new_request[3] and new_request[2] both do not equal 1 (“else”), but new_request[1] equals 1, then temp_out[1] equals 0 and temp_out[0] equals 1:



And finally, according to Figure 16, if none of the new_requests equals 1, then temp_out[1] and temp_out[0] both equal 0. Again, the schematic confirms as much:



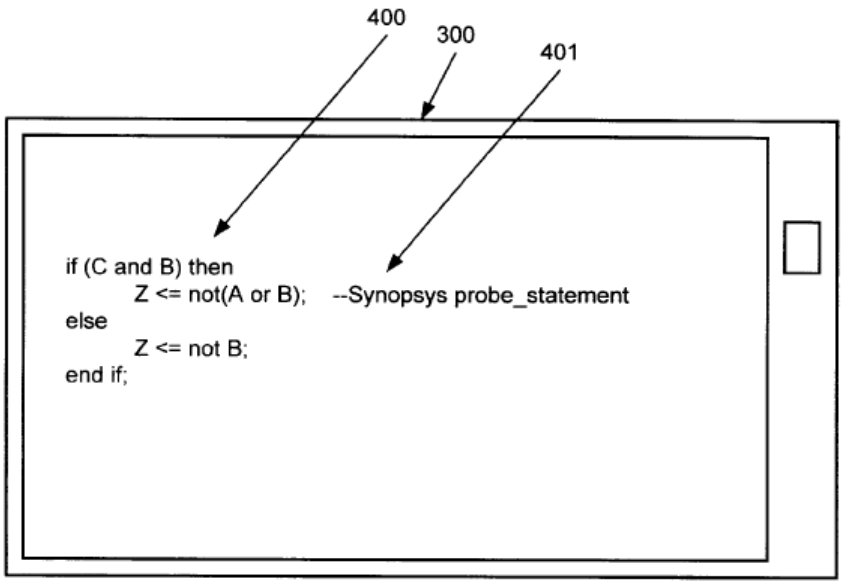
Therefore, by examining the only four possible (and unique) results of the temporary outputs in Figure 16 (as well as Figure 18), the designer knows which line in the if-then-else statement was executed. If both temporary outputs are 1, then the first instruction line of source code was executed. If only temp_out[1] is 1, then the second instruction was executed. If only temp_out[0] is 1, then the third instruction was executed. If neither temporary output is 1, then the fourth instruction line was executed.

In short, the values of the temporary output signals—which were created by instrumenting the source code using Gregory's method—indicate which of the "if-then" or "else" statements was performed and, therefore, the "execution status" of "at least one" of the source code

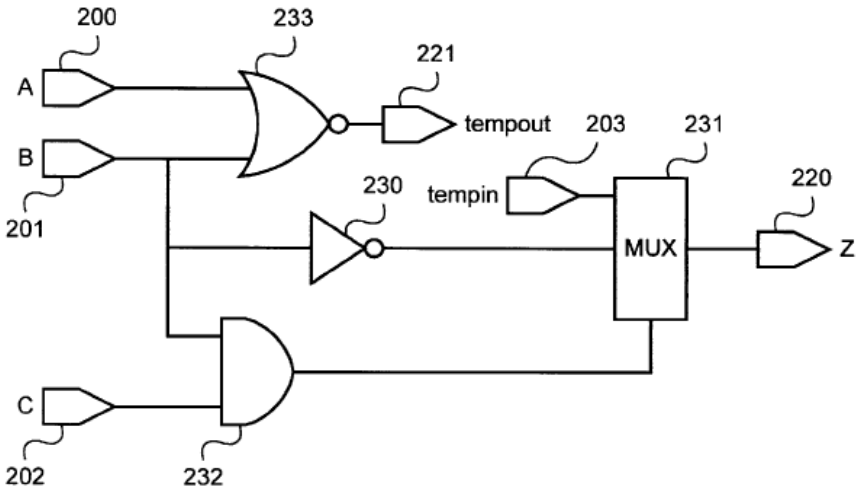
"statements" identified and instrumented. Because the Board failed to address this argument—and given that these figures in Gregory plainly are an anticipatory disclosure—the final written decision must be vacated and remanded.

b. Figures 8 and 9 disclose execution status.

Gregory also discloses "execution status" through Figures 8 and 9. As discussed above (at 6-8, 18-20), they disclose a straightforward "if-then-else" statement with instrumentation in the form of a probe statement.



A1112 Fig. 8. Figure 9 represents that same source code graphically:



A1113. In two different ways, these figures demonstrate that Gregory discloses "execution status" by providing "information regarding whether a particular HDL instruction has been performed." A68 (final written decision). As to each of these disclosures, the Board erred in its treatment of the reference, and also erred more fundamentally and as a matter of law in the fashion in which it undertook its analysis.

i. Analyzing the value of "tempout 221" (which equates to "not(A or B)" in the source code) can enable a designer to determine a broad range of information: the value of B; whether the condition "if (C and B)" is satisfied; which instruction line ("then" or "else") in the source code was followed; and the value of the output (Z). In short, tempout 221 gives an output value that can indicate which statement was executed, and thereby disclose execution status.

To explain, if "tempout" equals 1, then A and B must both have a value of 0.[13] Knowing that A and B both have a value of 0, the designer also knows that the condition "if (C and B)" is not satisfied.[14] Because "if (C and B)" is not satisfied, the code specifies that it is the "else" branch of the source code that is active. As a result, the output (Z) must be 1 (because, under the "else" branch, Z is "not B," and B equals 0).

The Board, however, determined that Gregory does not teach "execution status." It reasoned that "tempout" in Figures 8 and 9 does not disclose execution status because (according to Mentor's expert) "tempout" in Figure 9 only "reflects the result of [the logical operation] 'not(A or B),'" but "does not indicate" "whether the 'if' condition [if (C and B)] is true." A72 (discussing A1905 ¶¶ 73-74(Mentor's Expert's Decl.), which asserts that "[k]nowing just the state of 'not(A or B)' provides *no information* as to whether the statement is executed because the condition '(C and B)' was true" (emphasis added)). That may

---

[13] This is because 233 is a NOR gate, and if either of its inputs has a value of 1, the output will be 0. Since the output (tempout) is 1, inputs A and B must be 0. *Supra* at 12.

[14] This is because B has a value of 0, whereas B and C must both equal 1 for "if (C and B)" to be satisfied.

be in the circumstance where “tempout” has a value of 0.[15] But the Board’s broad conclusion—that “tempout” “does not indicate” “whether the ‘if’ condition [if (C and B)] is true”—is simply and flatly mistaken. As explained immediately above, if the value of “tempout” is 1, then the value of B must be 0. Knowing that the value of “tempout” is 1, and that the value of B therefore is 0, “reflects” far more information than merely “the result of ‘not(A or B)” in gate 233, A72—and Mentor has not argued otherwise. Knowing that the value of B is 0 means knowing that the condition “if (C and B)” is not true, that the “else” line in the source code of Figure 8 would execute, and that Z equals 1 because “Z <= not(B).” Therefore, by the Board’s own terms, “tempout” indicates the code’s “execution status” by demonstrating “whether the ‘if” condit-ion is true or not.” A72. Gregory therefore anticipates.

The Board refused to accept this explanation because “Synopsys … d[id] not point to any expert testimony” explaining how Gregory discloses “execution status.” *Id.* This error in approach

---

[15] In that circumstance, the designer would not necessarily know the value of B because an output of 0 for a NOR gate (such as 233) could be the result of either input (A or B) having a value of 1, so the designer cannot know with certainty the value of B.

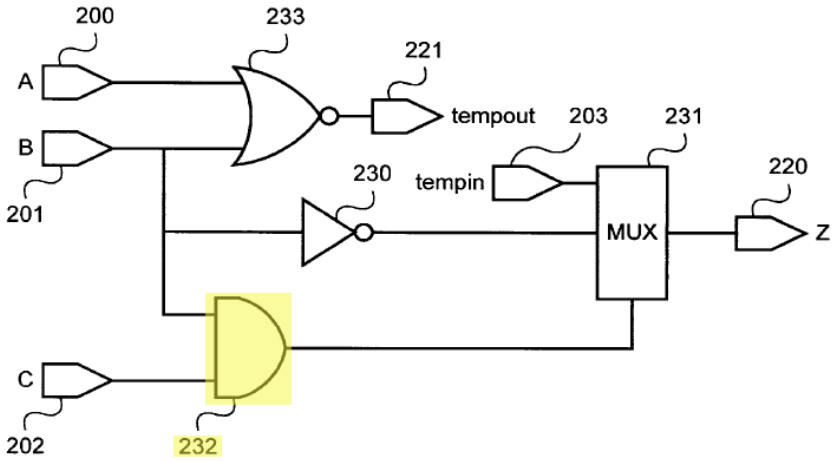
requires that the Board decision be vacated. Contrary to the Board's insistence on expert testimony, "there is no invariable requirement that a prior art reference be accompanied by expert testimony" in order to anticipate. *Meyer Intellect'l Props.*, 690 F.3d at 1374 (quotation marks and brackets omitted); *In re NTP, Inc.*, 654 F.3d 1279, 1301 (Fed. Cir. 2011) (a prior-art "reference itself provides the necessary evidence" to demonstrate anticipation); *Am. Calcar*, 651 F.3d at 1342 (same). The Board is comprised of judges with technical training, 35 U.S.C. § 6, and is constituted to evaluate technical arguments about technical material. *See* Joe Matal, *A Guide to the Legislative History of the America Invents Act: Part II of II*, 21 Fed. Cir. Bar J. 540, 601 (2012) (post-grant reviews conducted within the PTO because it provides "a cost-effective and technically sophisticated environment" to "mak[e] validity determinations" (quotation marks omitted)). The Board's job is to assess the prior art and to determine who has the better arguments, not to count who has more experts. This is especially so given the highly streamlined nature of *inter partes* review, which does not require expert testimony. *See* 35 U.S.C. § 312(a)(3)(B) (contemplating expert affidavits or declarations only "if the petitioner relies on expert opinions").

If the Board had engaged our arguments on their merits, that would be one thing. But by simply parroting the testimony of Mentor's expert and calling Synopsys' argument "unsupported" because it did "not point to any expert testimony," A72-73, the Board failed to adequately address and respond to Synopsys' arguments. This was error. *See Iowa*, 218 F.3d at 759; *BNP Paribas Energy Trading GP v. FERC*, 743 F.3d 264, 269-70 (D.C. Cir. 2014) ("opaque" and "out-of-hand" "dismissal" of arguments "falls well short" of an agency's obligation to "provide an adequate explanation" of its decisions (quotation marks omitted)); *PSEG Energy Resources & Trade LLC v. FERC*, 360 F.3d 200, 204 (D.C. Cir. 2004) ("two sentence[]" agency response to an argument is "wholly inadequate").

The Board's duty to adequately address the arguments presented to it derives from an agency's fundamental obligation to make "[n]ecessary findings … expressed with sufficient particularity to enable our court, without resort to speculation, to understand the reasoning of the Board, and to determine whether it applied the law correctly and whether the evidence supported the underlying and ultimate fact findings." *Gechter*, 116 F.3d at 1457; *accord Lee*, 277 F.3d at 1342; *see*

also SEC v. Chenery Corp., 318 U.S. 80, 94 (1943) ("[T]he orderly functioning of the process of review requires that the grounds upon which the administrative agency acted by are clearly disclosed and adequately sustained"). And here, through its cursory rejection of Synopsys' arguments for want of expert opinion, the Board failed to demonstrate that "the evidence supported" its decision. *Gechter*, 116 F.3d at 1457. This "'is grounds for striking down [its] action.'" *Lee*, 277 F.3d at 1344 (quoting *Mullins v. Dep't of Energy*, 50 F.3d 990, 992 (Fed. Cir. 1995)).

ii. We said there are two ways that Gregory Figures 8 and 9 disclose "execution status." The first was the value of "tempout" in Figure 9 discussed above. In addition, execution status is disclosed through the output of the AND gate (232), which is highlighted below:



A1113 Fig. 9 (Gregory '109). As the schematic shows, AND Gate 232 receives inputs B and C, and it therefore represents the "if" condition in the statement "if (C and B) then…." One can look at the output of this gate to determine which statement was executed. If the output of that gate is 1, that means (by definition) that the condition "(C and B)" is true, and the "if" condition therefore was satisfied. If instead the output of gate 232 is 0, then "(C and B)" is false, and the "if" condition was not satisfied. This is sufficient to show "execution status" under the Board's own reasoning, because knowing "whether the 'if' condition is true or not" indicates the "execution status of the HDL statement." A72 (final written decision).

The Board did not address this argument on its merits, and its decision not to do so requires vacatur and remand so that this argument may be considered by the Board in the first instance. According to the Board, "the argument is presented for the first time in Synopsys's reply and is not responsive to arguments made in Mentor Graphics's response." A74 (citation omitted). On the contrary, Synopsys fully preserved its argument concerning Figures 8 and 9 by making this

argument at the first moment it made sense to do so, in direct response to Mentor's proposed claim construction.

When Synopsys petitioned for review, it relied upon Figures 8 and 9 in arguing that Gregory is anticipatory. *See, e.g.*, A169-70, 180-81. Mentor's response to Synopsys' petition (called the "preliminary response") did not contest that Gregory discloses "execution status." A271-73, 277 (discussing claims 1 and 28). The Board then instituted proceedings.

Thereafter, in its next paper, Mentor *for the first time* addressed whether Gregory teaches "execution status"; proposed a construction of "execution status"; and argued that Figure 9 does not disclose "execution status" under its proposed construction because, Mentor claimed, Figure 9 does not indicate whether the condition "if (C and B)" has been satisfied.[16] *In response*, Synopsys explained that it is unnecessary to determine whether the "C and B condition" is met to determine execution status, but that "[e]ven if it were necessary" to do so in order "to

---

[16] *See* A440, 450-52 (Patent Owner Response) (citing A1887, 1905 ¶¶ 41, 74(Mentor's Expert's Decl.), and arguing that "'tempout' … does not depend on the condition (C and B)" and "is not indicative of execution status … because it lacks information about the condition")).

satisfy the 'execution status' requirement"—i.e., even if Mentor's claim construction were correct—Gregory still anticipates because of "[t]he output of AND gate 232 in Fig. 9." A562-63 (Synopsys Reply).

In short, Synopsys did precisely what the PTO's rules contemplate—when Mentor made new arguments in its Patent Owner Response, Synopsys addressed them in its Reply. *See* 37 C.F.R. § 42.23(b) ("A reply may only respond to arguments raised in the corresponding opposition or patent owner response"); *see Novosteel SA v. United States*, 284 F.3d 1261, 1274 (Fed. Cir. 2002) ("reply briefs *reply* to arguments made in the response brief"); *accord ACLU v. City of Las Vegas*, 333 F.3d 1092, 1106 n.14 (9th Cir. 2003) (considering reply arguments that were "reasonable response[s] to points made in the … answering brief"); 16AA Charles Alan Wright et al., *Federal Practice and Procedure* § 3974.3 (4th ed. 2008) ("in a reply brief," a party may "respond to arguments raised for the first time in the [opposing party's response] brief"). Synopsys' argument in its reply brief was not "new," and the Board erred in refusing to address it.

* * *

Gregory discloses a method of "synthesizing the source code into a gate-level netlist including at least one instrumentation signal … indicative of an execution status of the at least one statement." A126(15:2-8) ('376 patent). Accordingly, Gregory anticipates claims 1 and 28 of the '376 patent. In the alternative and at a minimum, the Board erred by applying the wrong law regarding anticipatory teachings and in refusing to consider key evidence and arguments offered by Synopsys based on an erroneous belief that expert testimony is required and a misplaced theory of waiver. The Board's determination must be vacated and remanded for a complete analysis.

II. THE BOARD ERRED AS A MATTER OF LAW BY FAILING TO ISSUE A FINAL WRITTEN DECISION WITH RESPECT TO EACH CLAIM CHALLENGED BY SYNOPSYS.

Synopsys petitioned for *inter partes* review of claims 1-15 and 20-33 of the '376 Patent. *Supra* at 24-25. The Board determined that the conditions for instituting proceedings were met, A1-41, and ultimately issued a final written decision, A42-94. That final written decision, however, covered only certain of the challenged claims—specifically, claims 1-9, 11, and 28-29—and did not address the patentability of the other challenged claims. *Id.* The Board's failure to include

its disposition of the other claims in its final written decision directly contravenes the clear requirements of 35 U.S.C. § 318(a), and the final written decision therefore must be vacated in relevant part and remanded.[17]

Section 318(a) is plain on its face: "If an inter partes review is instituted and not dismissed under this chapter, the Patent Trial and Appeal Board *shall* issue a final written decision with respect to the patentability of *any patent claim challenged by the petitioner* and any new claim added under section 316(d)" (emphases added). This provision is mandatory and expansive, and it does not authorize the

---

[17] Synopsys also has initiated suit under the APA in the Eastern District of Virginia to challenge the Board's regulation, policy, and practice of granting *inter partes* petitions in part and then declining to deal with each challenged claim when it issues final written decisions. *Supra* at xiv-xv. The government has moved to dismiss, arguing that this Court has exclusive jurisdiction over challenges to the Board's final written decisions. At the same time, however, the government has suggested that this Court also may lack jurisdiction. *See* Defs.' Resp. to SSA Institute's Amicus Br. at 3 n.1, *Synopsys, Inc. v. Lee*, No. 14-674 (E.D. Va. Sept. 18, 2014), ECF No. 40. Synopsys believes that this issue is appropriately reviewed in district court under the APA. But, given the government's position—and because the government's views on questions of appellate review under the AIA have sometimes shifted, *see, e.g.*, PTO Br. at 16 n.2, *Versata Devel. Group, Inc. v. Lee*, No. 2014-1145 (Fed. Cir. Apr. 28, 2014), ECF No. 55. Synopsys is proceeding in both forums until the question is resolved.

Board to pick and choose which claims to address in the final written decision. The ordinary meaning of "shall" is "must"—if the condition is satisfied, the Board must act. *See, e.g.*, *Nat'l Ass'n of Home Builders v. Defenders of Wildlife*, 551 U.S. 644, 661 (2007); *Miller v. French*, 530 U.S. 327, 337 (2000); *Sharp Elecs. Corp. v. McHugh*, 707 F.3d 1367, 1373 (Fed. Cir. 2013) ("'Shall' is 'mandatory' language"); *Atl. Sugar, Ltd. v. United States*, 744 F.2d 1556, 1560 (Fed. Cir. 1984).

What the Board must do is equally plain: "issue a final written decision with respect to" not just selected claims, but "*any* patent claim challenged by the petitioner." 35 U.S.C. § 318(a) (emphasis added). Simply put, "[t]he word 'any' is generally used in the sense of 'all' or 'every' and its meaning is most comprehensive." *Barsebäck Kraft AB v. United States*, 121 F.3d 1475, 1481 (Fed. Cir. 1997) (internal quotation marks omitted); *accord United States v. Gonzales,* 520 U.S. 1, 5 (1997) ("Read naturally, the word 'any' has an expansive meaning: … 'one or some indiscriminately of whatever kind'" (citing cases and quoting Webster's Third New International Dictionary 97 (1976))); *United States v. Rosenwasser*, 323 U.S. 360, 363 (1945) ("The use of the words 'each' and 'any' to modify 'employee,' … leaves no doubt as to the Congressional

intention to include all employees within the scope of the [Fair Labor Standards] Act unless specifically excluded.").

Thus, "the word 'any' necessarily includes 'all,'" and the only question that remains is "'all of what'?" *Micron Tech., Inc. v. United States*, 243 F.3d 1301, 1308 (Fed. Cir. 2001). And, on that score too, the statute is clear: The final written decision must address "any patent *claim challenged by the petitioner*." 35 U.S.C. § 318(a) (emphasis added). A "claim challenged by the petitioner" is one—as these terms plainly suggest—that the petitioner challenged, which is to say included in the petition seeking *inter partes* review. Indeed, this same phrase is used to define what a petitioner must include in the petition: The petition must "identif[y], in writing and with particularity, each *claim challenged*, the grounds on which the challenge to each claim is based, and the evidence that supports the grounds for the challenge to each claim." *Id.* § 312(a)(3) (emphasis added). Similarly, the Director may institute *inter partes* review if "the information presented in the petition … and any response filed … shows that there is a reasonable likelihood that the petitioner would prevail with respect to at least 1 of the *claims challenged in the petition*." *Id.* § 314(a) (emphasis added). The PTO

itself uses the term "claims challenged" to mean the claims that were asserted in the petition, not those on which proceedings were instituted. *See* USPTO's AIA Trial Roundtables at 9 (distinguishing between "claims challenged" and "claims instituted"), *available at* http://tinyurl.com/ownwbs4; *see also id.* at 13, 25, 46.[18]

Here, however, the final written decision did not address each of the challenged claims. Instead, the final written decision addressed only a subset of the challenged claims—those as to which the Board instituted *inter partes* review.[19] If Congress had wanted the Board to limit the final written decision to that subset of the claims, "it would have been easy enough for Congress to say so." *Empire HealthChoice Assur., Inc. v. McVeigh*, 547 U.S. 677, 696 (2006). For instance, it could have said that the Board "shall issue a final written decision with respect to the patentability of any patent claim for which the Director

---

[18] For this same reason, the Board's regulation authorizing "review to proceed on … *some* of the challenged claims," 37 C.F.R. § 42.108 (emphasis added), violates the express mandate of §§ 314(a) and 318(a), and is contrary to law.

[19] *See* A44 ("Synopsys has shown that claims 5, 8, and 9 are unpatentable. Synopsys, however, has not met its burden to show by a preponderance of the evidence that claims 1-4, 6, 7, 11, 28, and 29 are unpatentable"); A93 (same).

determined to institute *inter partes* review under section 314(a)." Instead, it unequivocally required the Board to address all of the challenged claims in the final written decision.[20]

Nor is there any practical impediment to the Board following the clear statutory command. The Board already opines on each "challenged claim" when it decides whether to grant a petition. Here, it issued a 40-page decision doing just that. *See generally* A1-41. At a minimum, therefore, it could incorporate that reasoning into the final written decision. If the PTO determines that such a process creates duplicative effort, then rather than frontload the analysis prior to institution, the Board could grant review upon determining that one claim

---

[20] Indeed, the prior statute included language that the AIA does not, which the PTO relied on in that context to justify considering only certain claims. 35 U.S.C. § 313 (2010) (if "the Director finds that a substantial new question of patentability affecting a claim of a patent is raised," the Director shall order "inter partes reexamination of the patent *for resolution of the question*" (emphasis added)); Notice of Clarification of Office Policy to Exercise Discretion in Reexamining Fewer Than All the Patent Claims, 1311 Off. Gaz. Pat. Office 197 (2006); *see Sony Computer Entm't Am. Inc. v. Dudas*, No. Civ. A. 1:05CV1447, 2006 U.S. Dist. LEXIS 36856, *27 (E.D. Va. May 22, 2006) ("Had Congress intended reexamination to extend to all claims under the patent, it would have left out the final phrase 'resolution of the question.'").

meets the threshold requirement. That is fully consistent with the statute, which authorizes the Board to institute review "*of the patent*," 35 U.S.C. § 311(a), so long as "there is a reasonable likelihood that the petitioner would prevail with respect to *at least 1 of the claims challenged in the petition*," *id.* § 314(a) (emphases added).

Such a procedure has much to recommend it. Because estoppel only applies to claims that "result[] in a final written decision," *id.* § 315(e)(2), a final written decision that fails to address all "claims challenged" will have limited estoppel effect. But Congress's goal was "to force a party to bring all of [its] claims in one forum … and therefore to eliminate the need to press any claims in other fora." 154 Cong. Rec. S9989 (daily ed. Sept. 27, 2008) (statement of Sen. Kyl).[21]

---

[21] *See also* 157 Cong. Rec. S1376 (daily ed. Mar. 8, 2011) (statement of Sen. Kyl) ("if an inter partes review is instituted while litigation is pending, that review will completely substitute for at least the patents-and-printed publication portion of the civil litigation"); *accord id.* at S1361; *see also* 110 S. Rep. No. 259 at 71 (Jan. 25, 2008) (intending to "limit unnecessary and counterproductive litigation costs" by "tak[ing] issues off the table that cannot be resurrected in subsequent litigation"); 153 Cong. Rec. H10280 (Sept. 7, 2007) (statement of Rep. Jackson-Lee) ("[T]he bill establishes a single opportunity for challeng[ing]" the patent and "prohibits a party from reasserting claims in court that it raised in post-grant review").

Issuing a final written decision as to all claims challenged also will minimize inconsistent results. Under its current practice, the Board often construes claim language during its initial evaluation whether to institute review but then may change its construction in the course of issuing its final written decision. The Board did just that here. *Compare* A6-10 (construing "instrumentation signal" in institution decision) *with* A59-68 (re-construing term in final written decision).

But, whatever procedures the Board ultimately may adopt, what remains clear is that the Board's treatment of each "challenged claim" must be incorporated into its final written decision. This clear statutory requirement enables meaningful appellate review of the Board's decisions on patentability. That is because it is the final written decision that gives rise to review in this Court. *See* 35 U.S.C. § 319 ("A party dissatisfied with the final written decision of the Patent Trial and Appeal Board under section 318(a) may appeal the decision …."); *id.* § 141(c)*; see St. Jude Medical v. Volcano Corp.*, 749 F.3d 1373, 1375-76 (Fed. Cir. 2014). Accordingly, the Board's failure to comply with § 318(a)—that is, its failure to address the patentability of

all challenged claims in the final written decision—effectively shields significant aspects of its decisionmaking from appellate review.

The decision of the Board should be vacated in relevant part, and the case remanded with instructions to comply with § 318(a).

CONCLUSION

For the foregoing reasons, the judgment of the Board upholding claims 1 and 28 of the '376 patent should be reversed or vacated; the Board's judgment upholding claims 2-4, 6-7, 10-15, 20-27, and 29-33 should be vacated; and the matter remanded for further proceedings.

Respectfully submitted,

/s/ Eric A. Shumsky

Eric A. Shumsky
Orrick, Herrington & Sutcliffe LLP
1152 15th Street, NW
Washington, DC 20005-1706
Telephone: (202) 339-8400
Facsimile: (202) 339-8500
eshumsky@orrick.com

Counsel for Appellant Synopsys, Inc.