

The opinion in support of the decision being entered today was not written for publication and is not binding precedent of the Board.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte EDWARD W. POWELL, DOUGLAS C. BUHLER,
and DAVID A. BURGOON

Appeal No. 2006-1595
Application No. 09/798,484

ON BRIEF

Before KRASS, JERRY SMITH, and MACDONALD Administrative Patent Judges.

JERRY SMITH, Administrative Patent Judge.

DECISION ON APPEAL

This is a decision on the appeal under 35 U.S.C. § 134 from the examiner's rejection of claims 1-23, which constitute all the claims pending in this application.

The disclosed invention pertains to a simulator that simulates hardware designs. Specifically, the invention automatically specifies the configuration of mixed-language models to be simulated. The mixed-language model comprises at least one model written in a source code language and a model written in a

hardware description language (HDL). The invention also automatically specifies an appropriate source code function library configuration depending on whether a stand-alone source code simulation or mixed-language simulation is performed.

Representative claim 1 is reproduced as follows:

1. An apparatus for automatically specifying the configuration of a mixed-language model to be simulated in a simulator, the mixed-language model comprising at least one model to be written in a source code language and at least one model written in a hardware description language (HDL), the apparatus comprising:

a first logic identifying hierarchy paths within the source code model;

a second logic identifying hierarchy paths within the source code model that correspond to hierarchy paths in the HDL model;

a third logic identifying connections within the source code model to be enabled or disabled; and

a fourth logic identifying portions of the source code model that are to be modeled by the source code model and portions of the source code model that are to be modeled by the HDL model.

The examiner relies on the following references:

Shinde et al. (Shinde)	5,493,507	Feb. 20, 1996
Parson	6,053,947	Apr. 25, 2000
Hellestrand et al. (Hellestrand)	6,263,302	Jul. 17, 2001 (filed Jan. 26, 2000)

Burgoon, David A., A Mixed-Language Simulator for Concurrent Engineering, IEEE, Mar. 1998 ("Burgoon").

Martinolle, Froncoise, A Procedural Language Interface for VHDL and its Typical Applications, IEEE, Mar. 1998 ("Martinolle").

The following rejections are on appeal before us:

1. Claims 1-23 stand rejected under 35 U.S.C. § 112, first paragraph, as containing subject matter which was not described in the specification in such a way as to reasonably convey to one skilled in the relevant art that the inventor(s), at the time the application was filed, had possession of the claimed invention.

2. Claims 1-23 stand rejected under 35 U.S.C. § 101 as being directed to non-statutory subject matter.

3. Claims 1-6, 12-20, and 22 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Burgoon in view of Parson.

4. Claims 7, 9-11, 21, and 23 stand rejected under 35 U.S.C. § 103(a) as being unpatentable over Burgoon in view of Hellestrand, Martinolle, and further in view of Shinde.

5. Claim 8 stands rejected under 35 U.S.C. § 103(a) as being unpatentable over Burgoon in view of Hellestrand, Martinolle, Shinde, and further in view of Parson.

Rather than repeat the arguments of appellants or the examiner, we make reference to the briefs and the answer for the respective details thereof.

OPINION

We have carefully considered the subject matter on appeal, the rejections advanced by the examiner and the evidence of obviousness relied upon by the examiner as support for the prior art rejections. We have, likewise, reviewed and

taken into consideration, in reaching our decision, the appellants' arguments set forth in the briefs along with the examiner's rationale in support of the rejections and arguments in rebuttal set forth in the examiner's answer.

It is our view, after consideration of the record before us, that the specification reasonably conveys to the skilled artisan that the inventors had possession of the claimed invention at the time the application was filed. We are also of the view that the claimed invention constitutes statutory subject matter under 35 U.S.C. § 101. Moreover, we conclude that the evidence relied upon and the level of skill in the particular art would not have suggested to one of ordinary skill in the art the obviousness of the invention as set forth in the claims on appeal. Accordingly, we reverse.

We consider first the examiner's rejection of claims 1-23 under 35 U.S.C. § 112, first paragraph. Regarding independent claims 1 and 7, the examiner contends that the apparatus claims do not recite any hardware elements, but merely logic statements or program logic [answer, page 6]. According to the examiner, such claims therefore recite "form over substance" with no tangible result [id.]. Regarding independent claims 1, 12, and 22, the examiner further contends that the specification does not describe what constitutes a hierarchy as claimed and how such a hierarchy is formed [answer, pages 6-10]. The examiner also alleges that the specification fails to describe any logic, procedure, or algorithm that is used to identify the hierarchy paths within the source code and HDL model [id.]. Regarding independent claims 7, 21, and 23, the examiner

contends that the specification does not describe (1) what constitutes a first or second configuration of the source code function library, and (2) what procedure or algorithm specifies or selects the first or second configuration of the library as claimed [id.].

Appellants note that the specification was previously amended to incorporate the entire text of claims 1-23 and therefore adequately conveys to the skilled artisan that appellants had possession of the invention at the time of filing [brief, page 3]. Appellants also argue that the examiner's allegation that the claims recite mere "form over substance" with no tangible result is irrelevant to the proper inquiry under 35 U.S.C. § 112, first paragraph [reply brief, pages 2 and 3]. Additionally, appellants note that the examiner failed to recognize or account for knowledge possessed by a person skilled in the art [reply brief, page 3]. Appellants further note that the examiner's position regarding the disclosure's inadequate written description is inconsistent with the examiner's obviousness rejections [reply brief, pages 2 and 3]. Finally, appellants argue that skilled artisans would understand what a hierarchical path means in context of a Verilog model as evidenced by numerous cited articles and internet links [reply brief, page 4]. The examiner responds that the specification did not refer to such articles nor were they incorporated by reference [answer, page 28].

We will not sustain the examiner's rejection under 35 U.S.C. § 112, first paragraph. At the outset, we note that written description issues under § 112 typically arise when a question exists whether the specification as originally filed

supports claims not originally in the application. That is, written description issues typically involve whether the original application adequately supports claims added after filing or whether material added to the specification constitutes new matter under 35 U.S.C. § 132. See MPEP § 2163(I). See also MPEP § 2163.03 (citing four situations where written description issues typically arise). Accordingly, a strong presumption exists that an adequate written description of the claimed invention is present when the application is filed. MPEP § 2163(I)(A) [emphasis added]. Nevertheless, a question of whether the specification provides an adequate written description may arise even for an original claim when an aspect of the claimed invention has not been described with sufficient particularity that one skilled in the art would recognize that the applicant had possession of the claimed invention [id.]. In view of the strong presumption that the written description of the originally-claimed invention is adequate, however, rejections of original claims for lack of written description should be rare. MPEP § 2163.03.

To satisfy the written description requirement, the specification must describe the claimed invention in sufficient detail that the skilled artisan can reasonably conclude that the inventor had possession of the claimed invention. See Moba, B.V. v. Diamond Automation, Inc., 325 F.3d 1306, 1319, 66 USPQ2d 1429, 1438 (Fed. Cir. 2003). Accordingly, “[t]he possession test requires assessment from the viewpoint of one of skill in the art.” Moba, 325 F.3d at 1320, 66 USPQ2d at 1439. For original claims, possession may be shown,

among other things, by (1) describing an actual reduction to practice; (2) clearly depicting the invention in detailed drawings; or (3) sufficiently describing relevant and identifying characteristics of the invention. MPEP § 2163(II)(A)(3)(a). The specification, however, need not describe in detail that which is conventional or well known to skilled artisans. That is, the written description can be adequate even if every nuance of the claims is not explicitly described in the specification [id.]. See also Vas-Cath, Inc. v. Mahurkar, 935 F.2d 1555, 1563, 19 USPQ2d 1111, 1116 (Fed. Cir. 1991). Moreover, software aspects of inventions may be described functionally [see MPEP § 2106(V)(B)(1)].

In this case, we find that the specification adequately describes the claimed invention with sufficient particularity that the skilled artisan would reasonably conclude that the inventors had possession of the claimed invention. Initially, we note that the text of claims 1-23 is expressly incorporated into the specification.¹ Although a verbatim correspondence between the claim language and the specification is arguably dispositive of the issue of possession, we find additional support in the specification for the claimed subject matter that describes the invention with sufficient particularity to evidence possession. For example, regarding the claimed limitation calling for the first logic identifying hierarchy paths within the source code model, the specification states:

Once the C model has been loaded, a C model builder function is called to build the C model hierarchy, as indicated by block 22. During the initialization phase of startup, the C model threads

¹ See Amendment filed Mar. 28, 2005.

and nodes are instantiated in order to build the C model hierarchy [specification, page 18].

Regarding the limitation calling for identifying hierarchy paths in the source code model that correspond to hierarchy paths in the HDL model, the specification states:

After the C model has been built, the Verilog “initialization”...blocks are executed, as indicated by block 23...Each GET and PUT call knows where it was called from in the Verilog model hierarchy. These calls use their respective hierarchy paths in the Verilog model to find the corresponding hierarchy paths in the C model [specification, page 19].

The limitations calling for (1) identifying connections within the source code model that are enabled or disabled, and (2) identifying portions of the source code model to be modeled by the source code and HDL models respectively are described as follows:

From [a] tagging process, the C model can determine which ports are active in the C model and which are not. Ports in the C model that are inactive correspond to portions of the C model that are to be simulated with the Verilog model. Ports in the C model that are active correspond to portions of the mixed-language model that will be simulated with the C model portions and the resulting signals will be processed and interfaced to the Verilog model portions by the CVI modules 10 [id.].

Furthermore, regarding claims 7, 21, and 23, we find ample support for the claimed subject matter in the specification on at least Page 3, lines 1-8, Page 17, line 36 – Page 18, line 9 (describing loading C code within a shared library for either stand-alone source code or mixed-language simulations and specifying the

shared library name for the C model that is loaded), and Page 28, lines 1-10 (distinguishing the C model portions for stand-alone and mixed-language simulation).

In view of this evidence as well as the passages noted by appellants,² we conclude that the specification adequately describes the claimed invention with sufficient particularity such that the skilled artisan -- a computer software engineer with substantial industry experience -- would reasonably conclude that the inventors had possession of the claimed invention. Accordingly, we will not sustain the examiner's rejection under 35 U.S.C. § 112, first paragraph.

We next consider the examiner's rejection of claims 1-23 under 35 U.S.C. § 101 as being directed to non-statutory subject matter. The examiner reiterates that the independent claims recite "form over substance" with no tangible result and are therefore non-statutory [answer, pages 11 and 12]. Regarding independent apparatus claims 1 and 7, the examiner asserts that the recited logic elements "suggest only software components which do not add any statutory matter" [answer, page 11]. Regarding independent claims 22 and 23 (reciting a computer-readable medium), the examiner notes that because the specification did not state specifically what constitutes the claimed computer-readable medium, the medium could be, among other things, a carrier wave signal transmitting the computer program from a server to a client or via a network [answer, page 12]. The examiner essentially reasons that because (1)

² See Brief, pages 2 and 3.

carrier wave signals are non-statutory, and (2) nothing in the specification precludes claims 22 and 23 from encompassing such carrier waves, the claims must therefore be non-statutory [answer, page 12].

Appellants respond that computer software constitutes patentable subject matter, and the examiner's conclusion that the claims recite "form over substance" is an improper basis for rejecting claims under § 101 [brief, page 4; reply brief, page 5]. Appellants further note that the preambles of independent claims 1 and 7 recite practical applications of the claimed subject matter [brief, page 5]. Regarding claims 22 and 23, appellants argue that a computer-readable medium constitutes statutory subject matter and cite numerous patents claiming such a medium [brief, pages 6 and 7]. Appellants emphasize that merely because a claim term is not explicitly defined in the specification does not permit the examiner to adopt any definition that the examiner chooses. Rather, claim terms must be supported by -- and consistent with -- the specification [brief, page 8]. In this regard, appellants note that the specification does not support a carrier wave embodiment [reply brief, page 5].

We will not sustain the examiner's rejection of claims 1-23 under 35 U.S.C. § 101. The express language of 35 U.S.C. § 101 defines four categories of patentable subject matter: (1) processes; (2) machines; (3) manufactures; and (4) compositions of matter. The statute includes essentially "anything under the sun made by man." Diamond v. Chakrabarty, 447 U.S. 303, 308-09, 206 USPQ 193, 197 (1980). However, certain judicially-recognized exceptions -- namely

laws of nature, abstract ideas, and natural phenomena -- are not eligible for patent protection. Diamond v. Diehr, 450 U.S. 175, 185, 209 USPQ 1, 7 (1981).

On the other hand, practical applications of laws of nature, abstract ideas, or natural phenomena are eligible for patent protection. Diehr, 450 U.S. at 187, 209 USPQ at 8. A practical application of a judicial exception exists if the claim (1) transforms or reduces an article to a different state or thing, or (2) produces a useful, concrete, and tangible result. Diehr, 450 U.S. at 183, 209 USPQ at 6; State Street Bank & Trust Co. v. Signature Financial Group Inc., 149 F.3d 1368, 1374, 47 USPQ2d 1596, 1601 (Fed. Cir. 1998).

In this case, the examiner asserts that the claimed invention does not set forth a tangible result and is merely “form over substance.”³ At the outset, we note that a tangible result is a “real-world” result -- not an abstract result. See Gottschalk v. Benson, 409 U.S. 63, 71-72, 175 USPQ 673, 676-77 (1972).

In our view, the independent claims, taken as a whole, positively recite statutory subject matter. Claim 1, for example, recites “[a]n apparatus for automatically specifying the configuration of a mixed-language model to be simulated in a simulator..., the apparatus comprising...” [emphasis added]. The preamble of the claim establishes that the claimed subject matter not only is an

³ Although we generally agree with appellants that merely alleging that claimed subject matter is “form over substance” is not itself a proper basis for determining eligible subject matter under 35 U.S.C. § 101 [see reply brief, page 5], the MPEP in fact uses such language in connection with analysis of computer-related inventions for patentability under § 101. See MPEP § 2106(IV)(B)(1) (“Merely claiming nonfunctional descriptive material stored in a computer-readable medium does not make it statutory. Such a result would exalt form over substance.” (emphasis added)). But the MPEP’s usage of this language pertains to claiming nonfunctional descriptive material stored in a computer-readable medium – a situation that is not germane to the claims on appeal.

apparatus, but also an apparatus that automatically performs a beneficial, tangible function – namely automatically specifying the configuration of a mixed-language model to be simulated in a simulator.

The apparatus is defined in the claim by four enumerated logic elements that perform unique identification functions that together facilitate the recited automatic functionality. Additionally, the recited four logic elements are not limited to solely software implementations, but rather could also comprise hardware components that interact with the software. But even if the logic elements were implemented solely in software, the claim would nevertheless comprise statutory subject matter since it produces a useful, concrete, and tangible result – namely automatically specifying the configuration of a mixed-language model to be simulated in a simulator. Significantly, the term “automatically” itself requires implementation by a machine, such as a computer. In short, the identification functions achieved by each of the four recited logic elements are hardly abstract ideas, but rather achieve useful, concrete, and tangible results that automatically specify the configuration of a mixed-language model to be simulated in a simulator. Claim 1 therefore constitutes statutory subject matter under 35 U.S.C. § 101.

Similar considerations apply for independent claim 7. The preamble of the claim establishes that the claimed subject matter not only is an apparatus, but also an apparatus that automatically performs a beneficial, tangible function – namely automatically specifying a source code function library configuration for

hardware modeling simulation. Moreover, the determination and specifying functions achieved by the two recited logic elements respectively are hardly abstract ideas, but rather achieve useful, concrete, and tangible results that automatically specify a source code function library configuration for hardware modeling simulation. Claim 7 therefore constitutes statutory subject matter under 35 U.S.C. § 101.

Similarly, independent method claims 12 and 21 recite statutory subject matter under 35 U.S.C. § 101. In short, the recited method steps achieve useful, concrete, and tangible results that automatically specify (1) the configuration of a mixed-language model to be simulated in a simulator, or (2) a source code function library configuration for hardware modeling simulation as noted previously.

Turning to independent claims 22 and 23 reciting a computer-readable medium with a computer program, we agree with appellants that the claims recite statutory subject matter under 35 U.S.C. § 101. We note at the outset that the four code segments recited in claim 22 and the two code segments recited in claim 23 constitute functional descriptive material (i.e., "...data structures and computer programs which impart functionality when employed as a computer component"). See MPEP § 2106(IV)(B)(1). "When functional descriptive material is recorded on some computer-readable medium it becomes structurally and functionally interrelated to the medium and will be statutory in most cases since use of technology permits the function of the descriptive material to be

realized” [id.; see also MPEP § 2106(IV)(B)(1)]. Therefore, the functional descriptive material recited in conjunction with the computer-readable medium claimed in claims 22 and 23 constitutes statutory subject matter. We disagree with the examiner that merely because the scope and breadth of the term “computer-readable medium” could possibly encompass a carrier wave and that the specification ostensibly did not preclude such a possibility, claims 22 and 23 are therefore non-statutory. As the appellants indicate, the specification hardly supports such an expansive interpretation. In short, the examiner’s analysis is merely speculative and is not consistent with the specification. Accordingly, we will not sustain the examiner’s rejection of claims 22 and 23 under 35 U.S.C. § 101.

Since we do not sustain the examiner's rejection of independent claims 1, 7, and 12 under 35 U.S.C. § 101, we likewise do not sustain the examiner's rejection of dependent claims 2-6, 8-11, and 13-20.

We next consider the examiner’s rejection of claims 1-6, 12-20, and 22 under 35 U.S.C. § 103(a) as being unpatentable over Burgoon in view of Parson. In rejecting claims under 35 U.S.C. § 103, it is incumbent upon the examiner to establish a factual basis to support the legal conclusion of obviousness. See In re Fine, 837 F.2d 1071, 1073, 5 USPQ2d 1596, 1598 (Fed. Cir. 1988). In so doing, the examiner is expected to make the factual determinations set forth in Graham v. John Deere Co., 383 U.S. 1, 17, 148 USPQ 459, 467 (1966). The examiner must articulate reasons for the examiner’s decision. In re Lee, 277

F.3d 1338, 1342, 61 USPQ2d 1430, 1434 (Fed. Cir. 2002). In particular, the examiner must show that there is a teaching, motivation, or suggestion of a motivation to combine references relied on as evidence of obviousness. Id. at 1343. The examiner cannot simply reach conclusions based on the examiner's own understanding or experience - or on his or her assessment of what would be basic knowledge or common sense. Rather, the examiner must point to some concrete evidence in the record in support of these findings. In re Zurko, 258 F.3d 1379, 1386, 59 USPQ2d 1693, 1697 (Fed. Cir. 2001). Thus the examiner must not only assure that the requisite findings are made, based on evidence of record, but must also explain the reasoning by which the findings are deemed to support the examiner's conclusion. However, a suggestion, teaching, or motivation to combine the relevant prior art teachings does not have to be found explicitly in the prior art, as the teaching, motivation, or suggestion may be implicit from the prior art as a whole, rather than expressly stated in the references. The test for an implicit showing is what the combined teachings, knowledge of one of ordinary skill in the art, and the nature of the problem to be solved as a whole would have suggested to those of ordinary skill in the art. In re Kahn, 441 F.3d 977, 987-88, 78 USPQ2d 1329, 1336 (Fed. Cir. 2006) citing In re Kotzab, 217 F.3d 1365, 1370, 55 USPQ2d 1313 (Fed. Cir. 2000). See also In re Thrift, 298 F. 3d 1357, 1363, 63 USPQ2d 2002, 2008 (Fed. Cir. 2002). These showings by the examiner are an essential part of complying with the burden of presenting a prima facie case of obviousness. Note In re Oetiker, 977 F.2d

1443, 1445, 24 USPQ2d 1443, 1444 (Fed. Cir. 1992). If that burden is met, the burden then shifts to the applicant to overcome the prima facie case with argument and/or evidence. Obviousness is then determined on the basis of the evidence as a whole and the relative persuasiveness of the arguments. See Id.; In re Hedges, 783 F.2d 1038, 1039, 228 USPQ 685, 686 (Fed. Cir. 1986); In re Piasecki, 745 F.2d 1468, 1472, 223 USPQ 785, 788 (Fed. Cir. 1984); and In re Rinehart, 531 F.2d 1048, 1052, 189 USPQ 143, 147 (CCPA 1976). Only those arguments actually made by appellants have been considered in this decision. Arguments which appellants could have made but chose not to make in the briefs have not been considered and are deemed to be waived [see 37 CFR § 41.37(c)(1)(vii)(2004)].

Regarding independent claims 1, 12, and 22, the examiner essentially finds that Burgoon teaches every claimed feature except for (1) the first logic identifying hierarchy paths within the source code model, and (2) the second logic identifying hierarchy paths within the source code model corresponding to the HDL model [answer, pages 13 and 14]. The examiner cites Parson as teaching such limitations and contends that it would have been obvious to the skilled artisan at the time of the invention to include the first and second logic elements as taught by Parson in the system of Burgoon to allow a complete hierarchical design path in logging error messages and other messages [id.].

Appellants argue that the recited third and fourth logic elements are not disclosed by Burgoon as the examiner alleges [brief, pages 12 and 13].

Regarding the third logic limitation, appellants emphasize that Burgoon does not teach nor suggest a third logic that identifies connections within the source code model to be enabled or disabled [brief, page 12; emphasis in original]. The examiner responds that Burgoon's initialization functions that set the interface pointers, installing the packet handler module, and calling the initialization functions after the simulation controller determines which functions should be in C and Verilog respectively on Page 5 of Burgoon correspond to the third logic limitation [answer, page 31]. Regarding the fourth logic limitation, the examiner contends that Burgoon's disclosure on Page 3 of "read[ing] the initialization files that define which models are represented in C and which are described in Verilog, and configur[ing] the simulator accordingly" corresponds to the fourth logic limitation [answer, pages 32 and 33].

Appellants also argue that Parson does not disclose first logic identifying hierarchy paths within the source code model as claimed [brief, page 14]. Appellants further argue that Parson does not disclose second logic that identifies hierarchy paths within the source code model that corresponds to hierarchy paths in the HDL model [brief, page 15]. The examiner responds by citing various passages within the disclosure of Parson that teach hierarchical paths in the hardware, the source code model, the HDL model, and the correspondence among them [answer, pages 35 and 36]. The examiner also reiterates that the specification does not describe the logic, procedure, or

algorithm used to identify hierarchy paths within the source code model that correspond to hierarchy paths in the HDL model [answer, pages 36 and 37].

Appellants further argue that the examiner's combination of Parson and Burgoon is improper, particularly the examiner's motivation to combine the references [brief, pages 15-18]. The examiner responds that the skilled artisan interested in mixed-language simulation would be motivated to combine Burgoon and Parson since both references pertain to (1) simulating integrated circuits, and (2) source code and HDL simulation models and their correspondence [answer, page 38].

We will not sustain the examiner's rejection of independent claims 1, 12, and 22. Burgoon discloses a mixed-language simulator that can migrate models from the C world into a Verilog simulation [Burgoon, page 4; Fig. 3]. Such a migration includes: (1) a C interface module that provides versions of the model's interface functions to the Verilog world for evaluation; (2) a Verilog interface module that receives packets and manipulates signals in the Verilog world; and (3) a packet handler module that allows the Verilog world to generate requests for the C world [Burgoon, page 5].

We agree with appellants that Burgoon fails to reasonably teach or suggest third logic identifying connections within the source code model to be enabled or disabled as claimed. We disagree with the examiner that Burgoon's initialization functions setting the interface pointers, installing the packet handler module, and calling the initialization functions after the simulation controller

determines which functions should be in C and Verilog respectively on Page 5 of Burgoon reasonably correspond to the third logic limitation. Significantly, the claimed third logic limitation requires that the logic identify connections within the source code model to enable or disable [emphasis added]. Setting interface pointers and calling initialization functions after the controller determines which functions should be in C and Verilog respectively in Burgoon does not reasonably teach nor suggest identifying connections within the source code model to enable or disable as claimed. For at least this reason, we will not sustain the examiner's obviousness rejection.

Furthermore, the secondary reference, Parson, discloses a simulation model using object-oriented programming that, among other things, includes a C++ model constructor that creates a C++ model object hierarchy that is isomorphic to a corresponding hierarchical netlist structure [Parson, col. 7, lines 37-43]. Although Parson pertains to simulation modeling that incorporates the basic structure and notation as netlist languages, ultimately the model is an object-oriented program with aspects that are structurally equivalent to netlist languages [see, e.g., Parson, col. 3, lines 52-58]. Although Parson does teach maintaining subcircuit representation hierarchy as the examiner indicates, we see no reason why the skilled artisan would combine the isolated teachings of the object-oriented program of Parson with the mixed-language simulation of Burgoon. In short, we find no reasonable rationale or motivation to combine the references in the manner suggested by the examiner apart from hindsight

reconstruction of the claimed invention. And even if it were proper to combine the references, all claimed limitations have not been disclosed by the prior art as we indicated previously.

Accordingly, we will not sustain the examiner's obviousness rejection of independent claims 1, 12, and 22. Since we do not sustain the examiner's rejection of independent claims 1 and 12, we likewise do not sustain the examiner's rejection of dependent claims 2-6 and 13-20.

We next consider the examiner's rejection of Claims 7, 9-11, 21, and 23 under 35 U.S.C. § 103(a) as being unpatentable over Burgoon in view of Hellestrand, Martinolle, and further in view of Shinde. Regarding independent claims 7, 21, and 23, the examiner's rejection essentially finds that Burgoon teaches all claimed limitations except for (1) automatically specifying a source code function library configuration for hardware modeling simulation; (2) second logic for (a) specifying a first configuration of the library when a stand-alone source code simulation is to be performed, and (b) specifying a second configuration of the library when a mixed-language simulation is to be performed, where the different configurations enable the library to be used for stand-alone source code simulation and mixed language simulation [answer, pages 18-21].

The examiner cites Hellestrand as teaching an apparatus for automatically specifying a source code function library configuration for hardware modeling simulation. The examiner finds that it would have been obvious to the skilled artisan at the time of the invention to automatically specify a source code function

library configuration for hardware modeling simulation in the system of Burgoon to enable modeling the hardware in a higher level language [answer, page 19].

The examiner also cites Hellestrand as teaching second logic specifying a first configuration of the library when a stand-alone source code simulation is to be performed [answer, page 19]. The examiner also cites Martinolle as teaching implementing C functions that are impossible to perform in HDL [answer, page 19]. The examiner then concludes that it would have been obvious to the skilled artisan at the time of the invention to include second logic specifying a first configuration of the library when a stand-alone simulation is to be performed so that modeling could be implemented in a higher-level language such as C and enabling functions to be implemented that are otherwise impossible to implement in HDL [answer, pages 19 and 20].

The examiner further cites Shinde as teaching specifying a second configuration of the library when a mixed-language simulation is to be performed where different configurations enable the library to be used for stand-alone source code simulation and for mixed-language simulation [answer, page 20]. The examiner then finds that it would have been obvious to the skilled artisan at the time of the invention to include specifying a second configuration of the library when a mixed-language simulation is to be performed where different configurations enable the library to be used for stand-alone source code simulation and for mixed-language simulation to allow verifying the correctness of the hardware using multiple models [answer, page 20].

Appellants argue that there is no motivation to combine the references [brief, page 19]. Specifically, appellants contend that the examiner merely relied on the utility of the underlying claimed features as the motivation to combine the references [id.]. Appellants also argue that the cited prior art does not disclose the second logic limitation second logic for (1) specifying a first configuration of the library when a stand-alone source code simulation is to be performed, and (2) specifying a second configuration of the library when a mixed-language simulation is to be performed [brief, pages 19 and 20; emphasis added]. Appellants emphasize that such a limitation specifically recites an alternative configuration that is not taught nor suggested by the prior art references [brief, page 20]. Appellants note that the examiner cited two different references for each recited alternative – namely Hellestrand (stand-alone source code simulation) and Shinde (mixed-language simulation) [id.].

The examiner responds that the skilled artisan would take the teachings of Hellestrand and Shinde and the motivations provided by them to select mixed model simulation models with a first configuration of the library when a stand-alone source code simulation is to be performed and a second configuration of the library when a mixed-language simulation is to be performed [answer, pages 43 and 44]. The examiner further contends that because Shinde teaches multiple simulation models (i.e., functional model, language model, and structural model), the skilled artisan would select the appropriate library depending on which model is used [answer, page 45].

We will not sustain the examiner's obviousness rejection of independent claims 7, 21, and 23. We agree with appellants that there is no reasonable motivation for the skilled artisan to combine the isolated teachings of the four references in the manner suggested by the examiner essentially for the reasons noted by appellants. We add, however, that the examiner's reasons to combine the four references essentially asserts the very advantages that the claimed invention was designed to achieve. In our view, nothing in the references expressly or implicitly suggests achieving these advantages apart from appellants' own disclosure. Furthermore, we find no teaching or suggestion in the prior art to (1) determine whether a stand-alone source code simulation is to be performed, and (2) specify a first or second library configuration depending on the initial determination as claimed. In our view, the examiner has simply selected the claimed features admitted to be missing from Burgoon from the secondary references and reconstructed the claimed invention using only appellants' own disclosure as a blueprint. Therefore, we will not sustain the examiner's obviousness rejection of independent claims 7, 21, and 23. Since we do not sustain the examiner's rejection of independent claim 7, we likewise do not sustain the examiner's rejection of dependent claims 8-11.

In summary, we have not sustained the examiner's rejections with respect to any of the claims on appeal. Therefore, the decision of the examiner rejecting claims 1-23 is reversed.

REVERSED

ERROL A. KRASS)	
Administrative Patent Judge)	
)	
)	
)	
)	
JERRY SMITH)	BOARD OF PATENT
Administrative Patent Judge)	APPEALS AND
)	INTERFERENCES
)	
)	
ALLEN R. MACDONALD)	
Administrative Patent Judge)	

JS/ce

Appeal No. 2006-1595
Application No. 09/798,484

HEWLETT-PACKARD COMPANY
INTELLECTUAL PROPERTY ADMINISTRATION
P.O. BOX 272400
FORT COLLINS, CO 80527-2400