United States District Court,
E.D. Texas, Marshall Division.

BIAX CORPORATION,
v.
INTEL CORPORATION.

Civil Action No. 2:05-CV-184

March 1, 2007.

Eric M. Albritton, Attorney at Law, Longview, TX, Barry W. Graham, Darrel C. Karl, E. Robert Yoches, Edward J. Naidich, Griffith B. Price, Jr., Kenneth M. Frankel, Finnegan Henderson Farabow Garrett & Dunner, Washington, DC, Danny Lloyd Williams, Williams Morgan & Amerson, Houston, TX, Joseph E. Palys, Robert L. Burns, Finnegan Henderson Farabow Garrett & Dunner, Reston, VA, Roger D. Taylor, Finnegan Henderson Farabow Garrett & Dunner, Atlanta, GA, for Biax Corporation.

Eric Hugh Findlay, Ronald Stephen Vickery, Ramey & Flock, Robert M. Parker, Parker, Bunt & Ainsworth, P.C., Jennifer Parker Ainsworth, Wilson Sheehy Knowles Robertson & Cornelius PC, Tyler, TX, Alex Verbin Chachkes, Robert M. Isackson, Orrick Herrington & Sutcliffe, New York, NY, Christopher R. Ottenweller, Cynthia A. Wickstrom, G. Hopkins Guy, III, Ulysses Hui, Orrick Herrington & Sutcliffe, Menlo Park, CA, Harry Lee Gillam, Jr., Gillam & Smith, LLP, Marshall, TX, Janet Craycroft, Intel Corporation, Santa Clara, CA, Lisa Ward, Orrick Herrington & Sutcliffe, Irvine, CA, David J. Beck, Robert David Daniel, Beck Redden & Secrest, LLP, Houston, TX, Jeremy P. Oczek, Steven M. Bauer, Proskauer Rose LLP, Boston, MA, for Intel Corporation, et al.

MEMORANDUM OPINION AND ORDER

T. JOHN WARD**, United States District Judge.**

1. Introduction

Biax Corporation ("Biax") asserts various claims of five United States patents in this case. Four of the patents are asserted against both defendants, Intel Corporation ("Intel") and Analog Devices, Inc. ("ADI"). These are U.S. Patent Nos. 4,847,755 ("the '755 patent"); 5,021,945 ("the "5 patent"); 5,517,628 ("the '628 patent"); and 6,253,313 ("the '313 patent"). A fifth patent, U.S. Patent No. 5,765,037 ("the '037 patent"), is asserted only against defendant ADI. This opinion resolves the parties' various claim construction disputes. The court will address briefly the technology at issue in the case, then turn to the merits of the claim construction issues.

2. Background of the Technology

The patents are directed to a parallel processor computer system and methods. Specialized software, referred to as TOLL software, analyzes the output of a conventional compiler and adds intelligence to the

instructions to facilitate the processing of the instructions. Instructions that are mutually independent from one another and can be processed at the same time, or in parallel, are referred to as "naturally concurrent" instructions.

The TOLL software assigns information to the compiler output. In the preferred embodiment, the information includes an "instruction firing time," a "logical processor number" and "shared context storage management information." The instruction firing time ("IFT") identifies the time that the instruction will be executed by the processors. The logical processor number ("LPN") identifies the processor that will execute the instruction. The shared context storage management ("SCSM") information identifies information concerning the context file and the register level at which the program is operating.

In addition to software, the patents disclose hardware for use with the system. Figure 6 of the patents illustrates these hardware components at a high level. The components are all interconnected through full-access networks. The components include memory resources, logical resource drivers, processor elements, and shared context storage files. In the preferred embodiment, the processor elements are all identical, and they execute all of the instructions, except branch instructions. A branch instruction changes the sequential instruction flow in a program by jumping to another portion in the programs. A separate branch execution unit, or BEU, handles these types of instructions. The use of the specialized software, in conjunction with the hardware features of the system, facilitates the parallel processing of instructions. Bearing this background in mind, the court will address the claim construction disputes.

3. Discussion

A. General Principles Governing Claim Construction

"A claim in a patent provides the metes and bounds of the right which the patent confers on the patentee to exclude others from making, using or selling the protected invention." Burke, Inc. v. Bruno Indep. Living Aids, Inc., 183 F.3d 1334, 1340 (Fed.Cir.1999). Claim construction is an issue of law for the court to decide. Markman v. Westview Instruments, Inc., 52 F.3d 967, 970-71 (Fed.Cir.1995) (en banc), *aff'd,* 517 U.S. 370, 116 S.Ct. 1384, 134 L.Ed.2d 577 (1996).

To ascertain the meaning of claims, the court looks to three primary sources: the claims, the specification, and the prosecution history. Markman, 52 F.3d at 979. Under the patent law, the specification must contain a written description of the invention that enables one of ordinary skill in the art to make and use the invention. A patent's claims must be read in view of the specification, of which they are a part. *Id.* For claim construction purposes, the description may act as a sort of dictionary, which explains the invention and may define terms used in the claims. *Id.* "One purpose for examining the specification is to determine if the patentee has limited the scope of the claims." Watts v. XL Sys., Inc., 232 F.3d 877, 882 (Fed.Cir.2000).

Nonetheless, it is the function of the claims, not the specification, to set forth the limits of the patentee's claims. Otherwise, there would be no need for claims. SRI Int'l v. Matsushita Elec. Corp., 775 F.2d 1107, 1121 (Fed.Cir.1985) (en banc). The patentee is free to be his own lexicographer, but any special definition given to a word must be clearly set forth in the specification. Intellicall, Inc. v. Phonometrics, 952 F.2d 1384, 1388 (Fed.Cir.1992). And, although the specification may indicate that certain embodiments are preferred, particular embodiments appearing in the specification will not be read into the claims when the claim language is broader than the embodiments. Electro Med. Sys., S.A. v. Cooper Life Scis., Inc., 34 F.3d 1048, 1054 (Fed.Cir.1994).

This court's claim construction decision must be informed by the Federal Circuit's decision in Phillips v. AWH Corp., 415 F.3d 1303 (Fed.Cir.2005) (en banc). In *Phillips,* the court set forth several guideposts that courts should follow when construing claims. In particular, the court reiterated that "the *claims* of a patent define the invention to which the patentee is entitled the right to exclude." *Id*. at 1312 (emphasis added) (quoting Innova/Pure Water, Inc. v. Safari Water Filtration Sys., Inc., 381 F.3d 1111, 1115 (Fed.Cir.2004)). To that end, the words used in a claim are generally given their ordinary and customary meaning. *Id*. The ordinary and customary meaning of a claim term "is the meaning that the term would have to a person of ordinary skill in the art in question at the time of the invention, i.e. as of the effective filing date of the patent application." Id. at 1313. This principle of patent law flows naturally from the recognition that inventors are usually persons who are skilled in the field of the invention. The patent is addressed to and intended to be read by others skilled in the particular art. *Id*.

The primacy of claim terms notwithstanding, *Phillips* made clear that "the person of ordinary skill in the art is deemed to read the claim term not only in the context of the particular claim in which the disputed term appears, but in the context of the entire patent, including the specification." *Id*. Although the claims themselves may provide guidance as to the meaning of particular terms, those terms are part of "a fully integrated written instrument." *Id*. at 1315 (quoting Markman, 52 F.3d at 978). Thus, the *Phillips* court emphasized the specification as being the primary basis for construing the claims. *Id*. at 1314-17. As the Supreme Court stated long ago, "in case of doubt or ambiguity it is proper in all cases to refer back to the descriptive portions of the specification to aid in solving the doubt or in ascertaining the true intent and meaning of the language employed in the claims." Bates v. Coe, 98 U.S. 31, 38, 25 L.Ed. 68 (1878). In addressing the role of the specification, the *Phillips* court quoted with approval its earlier observations from Renishaw PLC v. Marposs Societa' per Azioni, 158 F.3d 1243, 1250 (Fed.Cir.1998):

Ultimately, the interpretation to be given a term can only be determined and confirmed with a full understanding of what the inventors actually invented and intended to envelop with the claim. The construction that stays true to the claim language and most naturally aligns with the patent's description of the invention will be, in the end, the correct construction.

Consequently, *Phillips* emphasized the important role the specification plays in the claim construction process.

The prosecution history also continues to play an important role in claim interpretation. The prosecution history helps to demonstrate how the inventor and the PTO understood the patent. Phillips, 415 F.3d at 1317. Because the file history, however, "represents an ongoing negotiation between the PTO and the applicant," it may lack the clarity of the specification and thus be less useful in claim construction proceedings. *Id*. Nevertheless, the prosecution history is intrinsic evidence. That evidence is relevant to the determination of how the inventor understood the invention and whether the inventor limited the invention during prosecution by narrowing the scope of the claims.

Phillips rejected any claim construction approach that sacrificed the intrinsic record in favor of extrinsic evidence, such as dictionary definitions or expert testimony. The *en banc* court condemned the suggestion made by Tex. Digital Sys., Inc. v. Telegenix, Inc., 308 F.3d 1193 (Fed.Cir.2002), that a court should discern the ordinary meaning of the claim terms (through dictionaries or otherwise) before resorting to the specification for certain limited purposes. *Id*. at 1319-24. The approach suggested by *Tex. Digital*-the assignment of a limited role to the specification-was rejected as inconsistent with decisions holding the specification to be the best guide to the meaning of a disputed term. *Id*. at 1320-21. According to *Phillips,*

reliance on dictionary definitions at the expense of the specification had the effect of "focus[ing] the inquiry on the abstract meaning of words rather than on the meaning of the claim terms within the context of the patent." *Id*. at 1321. *Phillips* emphasized that the patent system is based on the proposition that the claims cover only the invented subject matter. *Id*. What is described in the claims flows from the statutory requirement imposed on the patentee to describe and particularly claim what he or she has invented. *Id*. The definitions found in dictionaries, however, often flow from the editors' objective of assembling all of the possible definitions for a word. *Id*. at 1321-22.

Phillips does not preclude all uses of dictionaries in claim construction proceedings. Instead, the court assigned dictionaries a role subordinate to the intrinsic record. In doing so, the court emphasized that claim construction issues are not resolved by any magic formula. The court did not impose any particular sequence of steps for a court to follow when it considers disputed claim language. *Id*. at 1323-25. Rather, *Phillips* held that a court must attach the appropriate weight to the intrinsic sources offered in support of a proposed claim construction, bearing in mind the general rule that the claims measure the scope of the patent grant.

B. Corrected Specification

What constitutes "the specification" is at issue in this case. The defendants contend that the patentee submitted a corrected specification during the prosecution of the patents, and that the corrected specification improperly introduced new matter into the application. They argue that the claim terms should be construed in light of the original, rather than the corrected, specification. The plaintiffs argue that the corrected specification made only minor typographical changes and did not introduce new matter.

The applicable statute states that the corrected version of the patent "shall have the same effect and operation in law on the trial of actions for causes thereafter arising as if the same had been originally issued in such corrected form." 35 U.S.C. s. 254. The court has not located a case in which a party urged a claim construction position by relying on language in an original, as opposed to a corrected, specification. Several district courts have held, however, that "determining whether a patentee introduced new matter during prosecution is not appropriate during claim construction." Commonwealth Scientific Industrial Research Org. v. Buffalo Technology (USA), Inc., 2006 WL 1233122 (E.D.Tex.2006) (Davis, J.); Pliant Corp. v. MSC Mktg. & Tech., Inc., 416 F.Supp.2d 632, 643 n. 5 (N.D.Ill.2006); Reiffin v. Microsoft, 64 U.S.P.Q.2d (BNA) 1107, 1112 (N.D.Cal.2002). At bottom, the defendants' position is that the patentee improperly introduced new matter in the corrected specification. Based on the statute and these decisions, the court will construe the disputed terms based on the corrected specification in the issued patent. The court now turns to those issues.

C. Specific terms in dispute

1. Means (160) for statically adding intelligence to each instruction in each of said plurality of basic blocks

This phrase appears in claims 11, 12, and 13 of the '755 patent. It is a means-plus-function limitation. The parties dispute the proper construction of the function as well as the proper identification of the structure. In particular, the parties dispute whether "statically adding intelligence" should be construed to mean adding information prior to program execution (as the plaintiff contends) or after compilation and prior to execution (as the defendants contend). The defendants argue that this is a means-plus-function limitation, and the only embodiment described in the patent has the TOLL software operating on the output from the

compiler.

The function is construed to mean "statically adding intelligence to each instruction of said plurality of basic blocks." The term "statically adding intelligence" means "adding information prior to program execution." *See* ' 755 patent, col, 2, ll. 42-46 (" 'Static allocations' " are performed once, prior to execution, whereas "dynamic allocations" are performed by the hardware whenever the program is executed or run. The present invention uses a static allocation strategy and provides task allocations for a given program after compilation and prior to execution."). The corresponding structure is the TOLL software operating on output from the compiler. The court includes in the corresponding structure the corresponding algorithms disclosed in Figs. 1, 8, 9, 10, and 11. FN1

FN1. Pursuant to the relevant statute, the claim is construed to cover the corresponding structure described in the specification and equivalents thereof. 35 U.S.C. s. 112 para. 6.

2. Means for processing said transferred instruction/means for processing each received instruction

This term is also governed by section 112 para. 6. It appears in claims 11-13 of the '755 patent. The dispute is over the description of the corresponding structure. The plaintiff defines the structure generically, by pointing to Fig. 6, PE 640. The defendants ask the court to identify the processing means to include Figs. 6, 7, 21 and the descriptions in the '755 patent at various columns. The defendants' added description would make it clear that the corresponding structure includes the identical context-free processor elements shown in the preferred embodiment of the patent. The defendants argue that the only embodiment shows identical processor elements, and under s. 112 para. 6, the patentee is bound by the narrow structure disclosed in the specification. The court agrees with the defendants. This claim limitation is drafted in means-plus-function form, and the court adopts the defendants' proposed corresponding structure.

3. Processor Element

The next term in dispute is "processor element." This term is not drafted in means-plus-function format and accordingly the plaintiffs are entitled to the ordinary meaning of the term, unrestricted by limitations in the preferred embodiment. The plaintiff contends that this term means "a device in a computer that executes instructions." The defendants argue that the term means "a microprocessor or CPU." Although the court agrees with the defendants that the term "processor element" connotes a device that does more than simply "execute instructions", the court disagrees that the patentee used the term "processor element" and microprocessor to mean the same thing. In addition, portions of the specification support the view that the processor element is capable of both interpreting and executing instructions. The court has considered the parties' briefing and is persuaded that the term "processor element" means "a device that is capable of interpreting and executing instructions."

4. Natural Concurrencies/determining natural concurrencies/determine the natural concurrencies

The next group of terms in dispute use the phrase "natural concurrencies." These terms appear in claims 11-13, 20-22, and 26 of the '755 patent as well as in claims 1, 3, 6, 15, 28, and 35-37 of the "5 patent. Biax argues that the term "natural concurrencies" means "attributes of instructions that enable the instructions to be executed in parallel with other instructions." Intel and ADI argue that the term means "mutually independent instructions that can be executed in parallel and do not rely on one another for any information nor do they share any hardware resources in other than a read only manner."

The specification resolves this claim construction dispute. In the specification, the patents state:

Clearly, mutually independent instructions can be executed in parallel and are termed 'naturally concurrent.' Instructions that are independent can be executed in parallel and do not rely on one another for any information nor do they share any hardware resources in other than a read only manner.

'755 patent, col. 9, ll. 16-18. In light of this passage from the specification, the court defines "natural concurrencies" to mean "mutually independent instructions that can be executed in parallel."

5. Condition code value/condition value

These terms appear in claims 1, 2, 9, 10-12, 16, 17, 20, 21, 23, 25, 26, 29, and 30 of the '628 patent as well as in claims 3, 6, 7, 10, 13, 17, 19, 22, 23, and 25 of the '313 patent. During the claim construction hearing, the plaintiff agreed with the defendants' construction of "condition code value." The court accordingly construes that term to mean "the summary of an executed instruction represented as a plurality of bits." With respect to the term "condition value," the court adopts the same construction, given that the parties' briefing suggests that the terms should be construed consistently. *See* Biax Opening Brief, at 59; Defendants' Response Brief, at 23, n. 4.

6. Instruction firing time

The next dispute relates to the term "instruction firing time." The term appears in various claims of the '755 and the "5 patents. In the patents, the TOLL software adds intelligence to the instruction stream. Some of that intelligence provides an "instruction firing time" or IFT. The parties dispute whether the IFT may be a relative time or whether it must mean a specific time. The court is persuaded that the plaintiff's view of this term is more appropriate. One passage from the specification supports the view that the instruction firing time may be a relative one: "[i]n the preferred embodiment of the invention, [the added intelligence] is the assignment of an instruction's execution time *relative* to the execution times of the other instructions in the stream." '755 patent, col. 10, ll. 28-31. After considering the parties' arguments on this point, the court concludes that the definition adopted by the ITC judge in Biax's collateral proceeding against Texas Instruments is correct. The court construes the term in this case to mean "intelligence used to indicate when a given instruction may be executed."

7. Logical Processor Number

The TOLL software adds a second type of intelligence to the instruction stream. This intelligence is the logical processor number, or "LPN." The "logical processor number" limitation appears in claims 11, 12, and 20-22 of the '755 patent as well as in claims 1, 3, 6, 11, 13, 15, 21, 23, 28, 36, and 37 of the "5 patent.

The plaintiff asks the court to construe the term to mean "information that is used to indicate which processor will execute the instruction." The defendants construe it to mean "the number added to an instruction, identifying the particular processor assigned to execute the instruction." Once again, the court concludes that the ITC court correctly construed this term. The plaintiff's definition does not account for the fact that the term in dispute specifically uses the word "number." As such, this court defines "logical processor number" to mean "a number correlated by the hardware to an actual physical processor element."

8. A multiprocessor system for processing a plurality of programs

The preamble of claim 37 of the '755 patent uses the phrase "a multiprocessor system for processing a plurality of programs of different users, said system comprising: ...." The defendants argue that the preamble provides antecedent basis for some of the claim terms and should be construed to limit the claim. The body of claim 37 of the '755 patent refers to "said programs," "the user context file for said one program," and a "plurality of processor elements for processing said programs." The defendants argue that the preamble also refers to a "multiprocessor system" and that the court should limit the claim to a computer system with two or more processors or CPUs.

The court rejects this argument. Certain terms in the preamble provide antecedent basis, but "multiprocessor system" is not one of them. As a result, the court declines to construe the phrase multiprocessor system.

9. Parallel processor system/method for parallel processing

The preambles of certain claims of the '755 patent recite "a parallel processor system for processing ...". The preambles of certain claims of the ' 945 patent recite "a method for parallel processing, in a system, natural concurrencies in streams of low level instructions ..."

This dispute also concerns the extent to which the language in the preambles limits the claims. The defendants argue that the claims refer to the preambles for antecedent basis, and the court should construe the claims to be limited to "parallel processor systems" or "methods for parallel processing" in accordance with the language of the preambles. The plaintiff argues that the preambles do not limit the claim and that no construction is required. The court agrees with the plaintiff. The body of the claims at issue do not rely on the relevant language of the preamble for antecedent basis. As such, the court declines to construe the terms "parallel processor system" and "method for parallel processing"

10. Dynamically adding information

Claim 37 of the '755 patent recites:

A plurality of logical resource drivers, each said logical resource driver being operative on at least one said programs for dynamically adding information during program execution to instructions of said one program, said information identifying at least the user context file for said one program ....

The next dispute is whether the "dynamically adding information" language requires the addition of specific types of information, beyond that set forth in the claim. Biax contends that this term requires no additional instruction. The defendants contend that "dynamically adding information" means "dynamically adding to each instruction during program execution information composed of (1) the context/LRD identifier that is issuing the instruction, (2) the current procedural level of the instruction, and (3) the process identifier of the current instruction stream."

To support their construction, the defendants point to the description of the preferred embodiment found in columns 34-35 of the patent. The claim language, however, is not limited to the preferred embodiment and in fact states that the information need only identify "the user context file for said one program." As a result, the court rejects the defendants' attempt to further limit this term.

11. Means (670) for connecting

The parties dispute the types of connections required by certain claims in the '755 patent. The parties have grouped these terms together and they appear in claims 11, 12, 13, and 26 of the '755 patent. To illustrate, claim 12 of the '755 patent sets forth the following limitation: "second means (670) for connecting each of said processor elements with any one of said plurality of shared storage resources, ..." Biax contends that the court should construe the corresponding structure as Fig. 6, P-E Context Network 670. Intel and ADI ask the court to construe the function as "to connect every processor element to every set of shared resources." So construed, Intel and ADI contend that the court should specifically identify the corresponding structure as the full-access, non-blocking network 670 set forth in the specification. The parties' dispute centers on whether the court should identify the specific networks disclosed in the patent as the corresponding structure.

The specification states:

The LRD-memory network 630, the PE-LRD network 650, and the PE-context file network 670 are full access networks that could be composed of conventional crossbar networks, omega networks, banyan networks, or the like. The networks are full access (non-blocking in space) so that, for example, any processor element 640 can access any register file or condition code storage in any context (as defined herein below) file 660.

Based on the language of the claims and the specification, the court construes the function as "connecting each of said processor elements with any one of said plurality of shared storage resources." The court identifies the corresponding structure as the full-access, non-blocking network 670 described in the specification. The patent discloses alternative ways of implementing such a network, and accordingly the network could be composed of a conventional crossbar network, omega network, banyan network, or the like.

12. Selectively connecting/connected/selectively interconnected

The parties have briefed a series of these terms and the dispute is whether the terms are drafted in means-plus-function form. To illustrate, claim 20 of the '755 patent requires "selectively connecting" stored instructions to the processor elements. The terms do not use the term "means," but the defendants ask the court to construe them as means-plus-function limitations and to incorporate the full-access, non-blocking networks described in the specification. This argument is rejected. The defendants have not overcome the presumption that the limitations are non-means-plus-function terms. No additional construction is required.

13. The user context file for said one program

Claim 37 of the '755 patent requires adding information "identifying at least the user context file for said one program." The parties dispute what is meant by the term "user context file." Biax contends that the term "context file" means "a set of registers" and the term "user context file" means "a set of registers for a user." Intel and ADI contend that the term "user context file" means "a physical register file, assigned to the user of one program, which contains the data used by and specific to that one program."

Although the description of the preferred embodiment supports the defendants' construction, the court disagrees that the limitations should be read into the claim language. The specific language in the specification does not require that the invention be limited to a system having a different context file for each user. *See* '755 patent, col. 17, ll. 60-63 ("In *an* MIMD system of the present invention as shown in FIG. 6, each context file would contain data from a different user executing a program.") (emphasis added).

Moreover, the specification counsels against a limitation that the context file must be specific to only one program. "The diagram of FIG. 6 represents an MIMD system wherein each context file 660 corresponds to *at least one* user program." '755 patent, col. 15, ll. 34-36. (emphasis added). As a result, the court adopts a construction similar to the one proposed by Biax. A user context file is "a set of registers assigned to a user." *See* '755 patent, col. 15, ll.34-36.

14. Users

Certain language in the preamble of claim 37 limits the claim because the preamble language provides antecedent basis. In particular, the language "plurality of programs of different users" provides antecedent basis for the term "said programs" recited in the body of claim 37. At issue is the definition of "user" in the phrase "programs of different users."

Biax contends that "user" includes both persons and programs. Intel and ADI contend that "user" must be limited to persons. Neither construction is proper. It is true that the cited portions of the specification relied on by the defendants support the view that "user" is different from "program." *See* ' 755 patent, col. 15, ll. 3-7 ("The TOLL software of the present invention can be used in connection with a number of simultaneously executing programs, each *program* being used by the same or different *users*.") (emphasis added). Nothing in the specification, however, expressly limits the term "user" to a human being using a program to the exclusion of a machine. As a result, the court holds that the term "user" includes "a person or a computer."

15. Means (160) receptive of said plurality of basic blocks for determining said natural concurrencies within said instruction stream for each of said basic blocks, said determining means further having means for adding timing and processor information to each instruction in response to said determined natural concurrencies so that all processing resources required by any given instruction are allocated in advance of execution

This limitation appears in claim 26 of the '755 patent. The term is drafted according to s. 112 para. 6. There are two functions: First, the claim requires a means for "determining said natural concurrencies." Second, that means is required to have a means for adding timing and processor information.

Biax contends that the structure corresponding to the means for determining is that portion of the algorithm described in the '755 patent, col. 9, ll. 7-31; col. 8, ll. 2-23 used to identify hardware resources used by the instructions and determining dependencies. As for the means for adding timing and processor information- Biax contends that the structure corresponding to this function is the computer system 160 shown in Fig. 1 which sets bits in the instruction to 1 or 0 to represent the added intelligence.

The defendants agree that the first function is determining natural concurrencies. They suggest, however, that the second function (adding timing and processor information) is the equivalent of adding an instruction firing time (IFT) and a logical processor number (LPM). So defined, they point to the TOLL software algorithms set forth in Figs. 1 and 8-11 that add the IFT and LPM to the compiler output.

After considering the arguments of the parties, the court concludes that the defendants are correct. With respect to this means-plus-function term, the patentee clearly linked the algorithms in Figs. 1 and 8-11 operating on compiler output to the function 'adding timing and processor information to each instruction in response to the determination of which instructions can be executed in parallel so that all the resources required for processing by any given instruction are allocated in advance of program execution." As a result, the court construes the corresponding structure as the computer system operating on compiler output running

the algorithms shown in Figs. 1 and 8-11.

16. Logical Resource Driver

The next term in dispute is the "logical resource driver" or LRD. This term appears in claims 11, 12, 13, and 37 of the '755 patent. Biax argues that the various claims define the LRD by describing the functions the LRD performs. According to Biax, the LRD is a unit that stores received instructions and delivers the instructions to processor elements. Intel and ADI define the term to mean "a hardware device assigned to a single user having a data cache, instruction selection support including an instruction cache, a plurality of instruction queues for delivering instructions to the processor element assigned to each logical processor number, a branch execution unit and an instruction cache address translation unit, and a data cache address translation unit."

Neither side's construction is entirely appropriate. The term "logical resource driver" does not have a specific meaning in this art. Resort to the specification is necessary. The specification states:

The logical resource drivers 620 are unique to the system architecture 600 of the present invention. Each illustrated LRD provides the data cache and instruction selection support for a single user (who is assigned a context file) on a timeshared basis. The LRDs receive execution sets from the various users wherein one or more execution sets for a context are stored on an LRD. The instructions within the basic blocks of the stored execution sets are stored in queues based on the previously assigned logical processor number. For example, if the system has 64 users and 8 LRDs, 8 users would share an individual LRD on a timeshared basis. The operating system determines which user is assigned to which LRD and for how long.

'755 patent, col. 15, ll. 45-59. A more detailed depiction of the LRD is found in Fig. 15 and the accompanying description of the preferred embodiment.

The patent states that the LRDs are "unique to the system architecture 600 of the present invention." As a result, the court construes the term "logical resource driver" as "a hardware device which provides a data cache and instruction selection support system for a given user. The LRDs receive execution sets, store the instructions, and deliver the instructions to the processor elements."

17. Said processor elements being connected

Claim 37 of the '755 patent requires a plurality of processor elements, "said processor elements being connected to said plurality of sets of shared resources." The question is whether this limitation is drafted according to s. 112 para. 6. The court holds it is not. The claim language does not recite the term "means," and Intel and ADI have not overcome the presumption attaching to the claim language.

18. Means for adding at least an instruction firing time (IFT) and a logical processor number (LPN).

Claim 28 of the "5 patent is drafted in means-plus-function format. The language of the claim requires a "means receptive of said plurality of basic blocks for determining said natural concurrencies ... said determining means further adding at least an instruction firing time (IFT) and a logical processor number (LPN)." Biax suggests that the corresponding structure is simply the computer processing system 160 and the TOLL software 110. The defendants, consistent with their arguments set forth above, contend that the court should specifically identify the TOLL software algorithms listed in Figs. 1 and 8-11, operating on compiler output. The court agrees with the defendants' arguments and includes as the corresponding

structure the computer system operating on compiler output and running the algorithms disclosed in Figs. 1 and 8-11 of the "5 patent.

19. Shared context storage mapping (SCSM) information

Claim 35 of the "5 patent recites the phrase "shared context storage mapping information." Biax contends that this term means "information identifying a context file." Intel and ADI argue that this term means "static and dynamic information attached to each instruction identifying the context file assigned to a given user and the register level assigned to a given instruction."

Once again, this term was not defined in the art as of 1985. As such, the court turns to the specification. The patent states:

The additional and required information comprises two components, a static and a dynamic component; and the information *is termed* "shared context storage mapping" (SCSM). The static information results from the compiler output and the TOLL software gleans the information from the compiler generated instruction stream and attaches the register information to the instruction prior to its being received by an LRD. The dynamic information is hardware attached to the instruction by the LRD prior to its issuance to the processors. This information is composed of the context/LRD identifier corresponding to the LRD issuing the instruction, the absolute value of the current procedural level of the instruction, the process identifier of the current instruction stream, and preferably the instruction status information that would normally be contained in the processors of a system having processors that are not context free.

"5 patent, col. 34, l. 61-col. 35, l. 10. Although the language appears in the description of the preferred embodiment, use of the phrase "is termed" suggests that the patentees were defining the term "shared context storage mapping information." The court accordingly defines "shared context storage information" as "static and dynamic information that identifies the register level for the instruction as well as the context file."

20. Instruction stream/basic blocks

The plaintiff defines "instruction stream" to mean "a stream of instructions." The defendants contend that the term means "a sequential stream of instructions output from a conventional compiler."

The defendants point to the specification for a definition of "instruction stream." The relevant portion states:

Processing system 160 operates on standard compiler output 100 which is typically object code or an intermediate object code such as 'p-code.' The output of a conventional compiler is a *sequential stream of object code instructions hereinafter referred to as the instruction stream*."

'755 patent, col. 6, l.66-col. 7, l.5. (emphasis added). Read in context, however, the court believes that the patentees intended in this passage to define instruction stream in the context of features of the preferred embodiment. Given that this is not a means-plus-function term, the court defines "instruction stream" to mean "a stream of computer instructions." It is not necessary for the stream of instructions to be a sequential stream of object code output from a conventional compiler.

ADI proposes a similar limitation to the term "basic block." Biax contends that the term "basic blocks" means "groups of contiguous instructions." ADI contends that the term means "compiler output consisting of

blocks of instructions with only one branch at the end of each block." The court adopts Biax's construction. One portion of the specification suggests that basic blocks can be formed in the compiler. The term is therefore not limited to compiler output. *E.g.*, '755 patent, col. 7, ll. 29-31 ("The TOLL software, in this illustrated embodiment of the present invention, is designed to operate on the formed basic blocks (BBs) which are created *within* a conventional compiler.").

21. Context free

The next term in dispute is "context free." This term appears in claims 11, 13, and 32 of the '755 patent and in claim 33 of the "5 patent. Claim 11 of the '755 patent is illustrative and recites "a plurality of individual context free processor elements (PEs)." The plaintiff defines a "context free processor element" to mean that "the processor element does not retain execution state information from execution of prior instructions and that execution state information is directly accessible by other processor elements." Intel and ADI argue that context free means "containing no execution state information from the execution of previous instructions."

In the original specification, the patentees included the following language:

The term "context free" means that the processor elements contain no state information, e.g., condition codes, registers, program status words, flags, etc.

The corrected specifications omit this language. Intel and ADI contend that the court should nevertheless use this definition of "context free" in the construction of claim 11.

The court disagrees. The applicable portion of the specification attached to the issued patent does not include the language relied upon by the defendants. During prosecution, the patentees described the "context free" concept of the invention and stated:

The present invention further executes the basic blocks containing the added intelligence on a system using a plurality of identical processor elements each of which is incapable of retaining execution state information from prior operations. Hence, all processor elements are context free and this feature is not found in any of the above references ....

(Petition to Make Special, attached to Defendants' Responsive Brief, Exh. W, p. 22).

The court accordingly defines "context free processor element" as "a processor element that retains no execution state information from prior operations."

22. Means including said logical resource drivers for connecting each of said processor elements with any one of said plurality of memory locations

This limitation appears in claim 11 of the '755 patent. It is drafted according to s. 112 para. 6. The court construes the claimed function as "connecting each of said processor elements with any one of said plurality of memory locations."

The plaintiff argues that the corresponding structure is shown in Fig. 15 as the Data Cache Interconnection Network 1590. The defendants argue that the claim language uses the phrase "including said logical resource drivers" and, as a result, includes the entire LRD structures and the associated networks 650 and 630 depicted in Fig. 6. In light of the language of the claim, the court concludes that the defendants' construction

is correct. The claim language specifically recites that the "means" includes the LRDs and further requires that the means perform the function of connecting the processor elements to the memory locations. As a consequence, the corresponding structure includes the full-access, non-blocking networks 650 and 630 depicted in Fig. 6, along with the LRDs 600.

23. Means (620) receiving said basic blocks ...

Claim 26 of the '755 patent includes the limitation "means (620) receiving said basic blocks (BBs) of instructions having said added timing and processor information for storing said received instructions." The parties dispute whether the means for storing includes the entire instruction selection section 1510 of LRD 620 or is limited to the PIQ buffer 1560. *See* Fig. 15. The language of claim reads "means (620) receiving said basic blocks (BBs) of instructions having said added timing and processor information for storing said received instructions." The defendants accordingly contend that the claim language requires the inclusion of all of the instruction selection section 1510.

The court disagrees. The relevant function is "storing said received instructions." The instruction selection section 1510 performs several functions other than simply storing the received instructions. The specification explains that "[t]he instruction selection portion 1510 of the LRD has three major functions: instruction caching, instruction queuing and branch execution....One purpose of the instruction portion 1510 is to receive execution sets from memory, place the sets into the caches 1522 and furnish the instructions within the sets, on an as needed basis, to the processor elements 640." '755 patent, col. 30, ll. 16-28. The specification further states: "[t]he instruction(s) are then delivered to the PIQ bus interface unit 1544 of the LRD 620 where it is routed to the appropriate PIQ buffers 1560 according to the logical processor number (LPN) contained in the extended intelligence that the TOLL software, in the illustrated embodiment, has attached to the instruction. The instructions in the PIQ buffer unit 1560 are buffered for assignment to the actual processor elements 640." '755 patent, col. 31, 26-33.

The specification suggests that the instructions could be stored in both the caches 0-3 and the PIQ buffer1560. As a result, the court would ordinarily identify both as alternative corresponding structure. The patentee explained during prosecution, however, that "[i]t is buffers 1560 which are storage means for storing instructions of a basic block according to the logical processor (number) which will execute that instruction." *See* Defendants' Response Brief, Exh. FF, at 44. In light of this statement, the court identifies the corresponding structure as the PIQ buffers 1560 as the means for storing.

24. Means (650) connecting said plurality of processor elements with said logical resource driver

This language and similar language appears in claims 11-13 and 26 of the ' 755 patent. The issue is the extent to which this means-plus-function term requires the full-access, non-blocking networks described in the patent. Consistent with its prior rulings, the court identifies the corresponding structure as the full-access, non-blocking network 650 described in the specification. Such a network could be implemented as a conventional crossbar network, omega network, banyan network, or the like.

25. Coupled

This term appears in claims 3 and 17 of the '313 patent. The plaintiff argues that this term needs no construction. The defendants contend that the term means "totally coupled," which is a variant of the position that the claims (even the non-means-plus-function claims) are limited a network that provides each processor full access to each storage register in a non-blocking manner. This position is rejected. The term

"coupled" means "connected, directly or indirectly."

26. Firing time information

This term is used in the '037 patent, asserted against only ADI. Biax contends this term means information pertaining to the time when an instruction is to be executed. ADI argues that it means a line of computer code providing the specific time when an instruction is to be executed. The court construes the term "firing time information" to mean "intelligence used to indicate when a given instruction may be executed." This construction is consistent with the court's prior construction of "instruction firing time."

27. Scheduling

This term appears in claims 8 and 9 of the '037 patent, asserted against ADI. In its brief, ADI asks the court to construe the term "scheduling" to mean "adding to each of the branch instructions computer code that assigns an instruction firing time." Biax argues that the term needs no construction and, in any event, the term "scheduling" is broader than adding an instruction firing time. The court agrees with Biax. Scheduling could also embrace the concept of re-ordering instructions. Accordingly, the court declines to limit the term as proposed by ADI. The term "scheduling" is entitled to its ordinary meaning and needs no construction.

28. Adding Information to Said Branch Instruction

This term appears in claim 7 of the '037 patent. ADI seeks a construction that would limit the information to code that assigns an instruction firing time. Claim 10, however, explicitly calls out adding instruction firing time. This term appears in claim 7 and accordingly is not limited. The specification supports the view that information other than instruction firing time may be added to the instruction. The court further rejects ADI's argument that the prosecution history requires the limitation proposed by ADI.

29. Instruction cycle/variable number of instruction cycles

This term appears in claims 1-12 of the '037 patent. The court construes this term as proposed by Biax. The term means "the period of time between the issuing of two successive instructions." No additional construction of "variable number of instruction cycles" is required.

30. Beginning execution of said branch instruction

This term appears in claims 7, 8, 9, and 12 of the '037 patent. The plaintiff argues that the execution of the branch instruction begins at the time it is sent to the branch execution unit. ADI argues that the execution of the instruction begins when it is loaded into the instruction register in the branch execution unit. The court agrees with this view. The claim language is drafted in terms of the "execution" of the branch instruction. Based on this language, the court concludes that execution of the instruction cannot begin until the branch instruction arrives at the branch execution unit. The execution of the branch instruction therefore begins when the instruction is loaded into the register in the BEU. The court construes the term to mean "loading the branch instruction into an instruction register in the branch execution unit."

31. Completing the execution of said branch instruction

This phrase appears in claims 1-12 of the '037 patent. Biax contends that this term means "delivering the target address of a branch instruction so that the target instruction will be the next instruction fetched for

execution." ADI contends that the term means "delivering the branch target address of the next basic block to the instruction cache control." The dispute is whether the delivery must be to the instruction cache control. The court agrees with Biax that ADI's construction improperly limits the claim to the preferred embodiment. The court adopts Biax's construction for this term.

32. Means ... for determining the branch instruction within each said basic block of said program[s], said determining means further adding firing time information to said branch instruction[s]

This term is drafted according to s. 112 para. 6. The corresponding structure is the computer system running the TOLL software algorithms disclosed in Figs. 1 and 8-11, operating on compiler output.

33. Means receptive of [each] said programs for determining the branch instruction within each of said basic block of [each of] said programs, said determining means further scheduling processing of said branch instructions.

These terms appear in claims 3 ad 4 of the '037 patent. Section 112 para. 6 applies. The corresponding structure is the computer system running the TOLL software algorithms disclosed in Figs. 1 and 8-11, operating on compiler output.

34. Means for processing non-branch instructions

This term is also drafted according to s. 112 para. 6. The claimed function is processing non-branch functions. The corresponding structure is the identical, context-free processor elements 640 disclosed in the specification.

35. Means ... for completing execution of said branch instructions no later than the same time as said processing means is processing the last to be executed non-branch instruction in said basic block so that the execution of said branch instruction occurs in parallel with the execution of said non-branch instructions in said basic block

This term is drafted according to s. 112 para. 6. The parties agree that corresponding structure includes delay unit 1940 and BEU 1548. ADI asks the court to include a delay field in instruction register 1900, connected to the Instruction Cache Control Unit 1518. The court rejects ADI's arguments and identifies the corresponding structure as the BEU 1548 and the delay unit 1940.

36. Means ... for beginning execution of said branch instruction

This term is also drafted according to s. 112 para. 6. The parties agree that BEU 1548 is a part of the corresponding structure. They dispute whether the court should also include instruction register 1900 and delay unit 1940. Biax agrees that instruction register 1900 should be included, but the delay unit 1940 is part of the means for completing execution (described above) and should not be included. The court agrees that delay unit 1940 is a part of the means for completing execution and accordingly does not include it as corresponding structure. The relevant claim language requires a means for beginning execution. The corresponding structure is BEU 1548 and instruction register 1900.

4. Conclusion

The court adopts the above definitions for those terms in need of construction. The court recognizes that it

has not construed all of the terms proposed in the parties' briefing. Nevertheless, the court has attempted to construe all of the terms that the parties addressed in oral argument as the terms that would allow the parties to resolve their disputes. The parties are ordered that they may not refer, directly or indirectly, to each other's claim construction positions in the presence of the jury. Likewise, the parties are ordered to refrain from mentioning any portion of this opinion, other than the actual definitions adopted by the court, in the presence of the jury. Any reference to claim construction proceedings is limited to informing the jury of the definitions adopted by the court.

E.D.Tex.,2007.
Biax Corp. v. Intel Corp.