

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE PATENT TRIAL AND APPEAL BOARD

INTEL CORPORATION,
Petitioner

v.

PACT XPP SCHWEIZ AG,
Patent Owner

**DECLARATION OF DR. CARL SECHEN UNDER 37 C.F.R. § 1.68 IN
SUPPORT OF PETITION FOR *INTER PARTES* REVIEW OF
U.S. PATENT NO. 9,075,605**

TABLE OF CONTENTS

	Page
I. Introduction.....	1
II. Background and Qualifications	5
III. Understanding of Patent Law	8
IV. Background.....	11
A. Background of the Field Relevant to the '605 Patent	11
B. Summary of the '605 Patent.....	20
C. Claims of the '605 Patent	22
D. Summary of the Prosecution History	23
V. Level of Ordinary Skill in the Pertinent Art.....	25
VI. Detailed Invalidity Analysis	26
VII. Background on Prior Art References	27
A. Background on Nicol.....	27
B. Background on Kling	32
C. Background on DeHon.....	35
D. Background on Bhatia.....	37
VIII. Claims 1-6 Are Obvious Based On Nicol In View Of Kling under patent owner's constructions	39
A. Reasons to Combine Nicol and Kling.....	39
B. Claim 1	42
C. Dependent Claim 2	66
D. Dependent Claim 3	68

E.	Dependent Claim 4	70
F.	Dependent Claim 5	71
G.	Dependent Claim 6	72
IX.	Claims 1 – 6 are obvious Based On Nicol in combination with Kling, and dehon under petitioner’s constructions	74
A.	Reasons to Combine Nicol, Kling, and DeHon	76
B.	Claim 1	82
X.	Claims 1–6 Are Obvious In View of Nicol in Combination with Kling, DeHon, and Bhatia Under Both Parties’ Constructions	87
XI.	Secondary Considerations of Non-Obviousness	90
XII.	Conclusion	92

I, Carl Sechen, Ph.D., do hereby declare as follows:

I. INTRODUCTION

1. I have been retained as an expert witness on behalf of Intel Corporation (“Intel”) for the above-captioned Petition for *Inter Partes* Review (“IPR”) of U.S. Patent No. 9,075,605 (“’605 patent”) (Ex. 1003). I am being compensated for my time in connection with this IPR at my standard consulting rate of \$375 per hour. My compensation is not affected by the outcome of this matter.
2. I have been asked to provide my opinions regarding whether or not Claims 1-6 of the ’605 patent (“the Challenged Claims”) would have been obvious to a person having ordinary skill in the art at the time of the alleged invention based on prior art.
3. The patent application which resulted in the ’605 Patent, Application No. 13/653,639, was filed on October 17, 2012. Ex. 1003 (’605 Patent), Cover. The ’605 patent claiming priority, through multiple divisional and continuation patents, to a foreign application filed March 5, 2001. *Id.*
4. The named inventors of the ’605 Patent are Martin Vorbach and Volker Baumgarte. *Id.* The original assignee of the ’605 Patent was PACT XPP TECHNOLOGIES AG. *Id.*

5. For the purposes of my Declaration, I have been asked to assume that the priority date of the alleged invention recited in the '605 Patent is March 5, 2001.
6. In preparing this Declaration, I have reviewed the '605 patent, the file histories of the '605 patent and U.S. Patent No. 9,552,047, numerous prior art references, and technical references from the time of the alleged invention.
7. I understand that claims in an IPR are generally given their ordinary and customary meaning, which 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.
8. I also understand that claim construction must take into account the intrinsic evidence of the patent, which includes the claims, specification, and prosecution history.
9. I am aware that the parties dispute certain terms at issue in their co-pending district court litigation. I am aware that the disputed terms are “data processing unit” and/or “data processing unit . . . adapted for sequentially processing data” and “to a minimum.” I am aware of both parties’ construction, but regardless of which construction applies, if any, it is my opinion that the prior art discussed in the IPR petition discloses the necessary limitations of the particular claim elements.

10. In forming the opinions expressed in this Declaration, I relied upon my education and experience in the relevant field of the art, and have considered the viewpoint of a person having ordinary skill in the relevant art. My opinions are based, at least in part, on the following references in view of the knowledge of one of ordinary skill in the art:

Reference	Date of Public Availability
U.S. Patent No. 9,075,605 (the “605 patent”) (Ex. 1003)	N/A
Certified File History of U.S. Patent Application No. 13/653,639 (“605 file history”) (without copies of the prior art references) (Ex. 1004)	N/A
Certified File History of U.S. Patent Application No. 14/219,945 (“047 file history”) (without copies of the prior art references) (Ex. 1008)	N/A
U.S. Patent No. 6,141,762 (“Nicol”) (Ex. 1009)	Filed: August 3, 1998 Issued: October 31, 2000
U.S. Patent No. 6,367,023 (“Kling”) (Ex. 1010)	Filed: December 23, 1998 Issued: April 2, 2002
U.S. Patent No. 6,535,798 (“Bhatia”) (Ex. 1011)	Filed: December 3, 1998 Issued: March 18, 2003
U.S. Patent No. 6,052,773 (“DeHon”) (Ex. 1012)	Filed: December 23, 1998 Issued: April 18, 2000

Reference	Date of Public Availability
Joint Claim Construction Chart, <i>XPP Schweiz AG v. Intel Corp.</i> , Case No. 1:19-cv-1006-UNA (D. Del.) (No. 67) (“Joint Claim Construction Chart”) (Ex. 1016)	N/A
U.S. Patent No. 6,704,877 (“Cline”) (Ex. 1018)	Filed: December 29, 2000 Issued: March 9, 2004
RUDOLF F. GRAF, MODERN DICTIONARY OF ELECTRONICS (7th ed. 1999) (“Graf”) (Ex. 1019)	Copyright date of 1999
<i>Advanced Configuration and Power Interface</i> (ACPI), Revision 1.0 (December 22, 1996) (Ex. 1020)	Published date of December, 1996
Michael Peter Kennedy & Leon O. Chua, <i>Hysteresis in Electronic Circuits: A Circuit Theorist’s Perspective</i> , in 19 INT. JOURNAL OF CIRCUIT THEORY AND APPLICATIONS, 471–515 (1991) (Ex. 1021)	October 1991, based on Ex. 1025 (Declaration of Pamela Stansbury)
John P. Hayes, <i>Computer Architecture and Organization</i> (3rd ed. 1998) (Ex. 1022)	Published in 1998
Kung, H.T., <i>Why Systolic Architectures?</i> , Computer Vol. 15 (Pub. January 1982) (Ex. 1014)	Published date of January 1982
U.S. Patent No. 6,397,343 (“Williams”) (Ex. 1026)	Filed: March 19, 1999 Issued: May 28, 2002

Reference	Date of Public Availability
I. Hong, D. Kirovski, Gang Qu, M. Potkonjak and M. B. Srivastava, <i>Power Optimization of Variable-Voltage Core-Based Systems</i> , in 18 IEEE TRANSACTIONS ON COMPUTER-AIDED DESIGN OF INTEGRATED CIRCUITS AND SYSTEMS, NO. 12, 1702-1714 (Dec. 1999) (“Hong”) (Ex. 1017)	Published date of December 1999

II. BACKGROUND AND QUALIFICATIONS

11. A copy of my *curriculum vitae* is presented as Exhibit 1002 and provides a comprehensive description of my relevant experience, including academic and employment history, publications, conference participation, and issued and pending U.S. patents.
12. I earned a B.E.E. in Electrical Engineering from the University of Minnesota in 1975, followed by a M.S. in Electrical Engineering from the Massachusetts Institute of Technology in 1977. I earned a Ph.D. in Electrical Engineering from the University of California, Berkeley in 1986.
13. I have been a Professor of Electrical Engineering for 33 years. Since August 15, 2005, I have been a Professor of Electrical and Computer Engineering at the University of Texas at Dallas. From July 1992 to August 14, 2005, I served as a Professor of Electrical Engineering at the University of Washington. From July 1986 through June 1992, I served as an Assistant

Professor and then Associate Professor of Electrical Engineering at Yale University.

14. Over these years my research has focused on the design and computer-aided design of digital integrated circuits, including computer architecture and the design of dynamic random-access memory (“DRAM”) and static random-access memory (SRAM) modules. I have taught numerous students how to design digital integrated circuits as part of the design courses that I have offered.
15. As a professor, I have developed and taught numerous courses, in particular, several courses that teach digital integrated circuit design and memory design in great detail. I have taught these courses continuously for the past 25 years. I have taught digital integrated circuit design and memory design to undergraduate and graduate students at the University of Washington and at the University of Texas at Dallas since 1995.
16. I have also been involved in numerous research projects on digital integrated circuit design and memory design. I have taught numerous graduate researchers how to design digital integrated circuits, including memories. Most of these had applications in computer architecture. I have also taught several Ph.D. graduate students how to design digital integrated circuits, including the design of various types of computer circuits and memories. I

have authored or coauthored over 190 papers and one book, the majority of which concern digital integrated circuit design and memory design.

17. I was elected a Fellow of the IEEE in 2002 for contributions to placement and routing of ASICs. IEEE stands for Institute of Electrical and Electronics Engineers, which is the leading professional association for electrical engineers. The Board of Directors of the IEEE awards the rank of “Fellow” to individuals with an extraordinary record of accomplishments in any of the IEEE fields of interest. The total number of IEEE members who can be named Fellows in any one year cannot exceed one-tenth of one percent of the total voting IEEE membership.

18. I received several research and teaching awards during my career. I received the Semiconductor Research Corporation’s Inventor’s Recognition Award in 1988 and in 2001. I also received the Technical Excellence Award from the Semiconductor Research Corporation in 1994. While serving as Professor at the University of Washington, I received the Outstanding Research Advisor award from the Department of Electrical Engineering in 2002. In 2008, I received the Distinguished Teacher of the Year Award from the Department of Electrical Engineering at the University of Texas at Dallas. I also received the Distinguished Teaching Award for the Erik Johnson School of Engineering and Computer Science in 2014.

19. Over the years, I have also received funding to conduct research in computer circuits and memory designs, including area-efficient and reliable embedded DRAM and SRAM design. Together with my graduate students, I have designed and fabricated various types of computational VLSI chips.
20. I am a co-inventor on two issued patents directed to transistor and computational logic technologies.

21. A copy of my CV is attached as Ex. 1002.

III. UNDERSTANDING OF PATENT LAW

22. Because I am an engineer and not an attorney, I have been provided with an understanding of the patent law relevant to conducting the analysis given in this report. The following represents my understanding of these issues.
23. I understand that prior art to the '605 patent includes patents and printed publications in the relevant art that predate the alleged priority date of the '605 patent.
24. I understand that a claim is invalid if it is anticipated or obvious. Anticipation of a claim requires that every element of a claim be disclosed expressly or inherently in a single prior art reference, arranged in the prior art reference as arranged in the claim. Obviousness of a claim requires that the claim be obvious from the perspective of a person having ordinary skill in the relevant art at the time the alleged invention was made. I understand that a

claim may be obvious in view of a single reference, or may be obvious from a combination of two or more prior art references.

25. I understand that an obviousness analysis requires an understanding of the scope and content of the prior art, any differences between the alleged invention and the prior art, and the level of ordinary skill in evaluating the pertinent art.

26. I also understand that the following factors are relevant to obviousness: (a) Combining prior art elements according to known methods to yield predictable results; (b) Simple substitution of one known element for another to obtain predictable results; (c) Use of known technique to improve similar devices (methods, or products) in the same way; (d) Applying a known technique to a known device (method, or product) ready for improvement to yield predictable results; (e) “Obvious to try” – choosing from a finite number of identified, predictable solutions, with a reasonable expectation of success; (f) Known work in one field of endeavor may prompt variations of it for use in either the same field or a different one based on design incentives or other market forces if the variations are predictable to one of ordinary skill in the art; (g) Some teaching, suggestion, or motivation in the prior art that would have led one of ordinary skill to modify the prior art

reference or to combine prior art reference teachings to arrive at the claimed invention.

27. I further understand that certain factors may support or rebut the obviousness of a claim. I understand that such secondary considerations include, among other things, commercial success of the alleged, patented invention, skepticism of those having ordinary skill in the art at the time of the alleged invention, unexpected results of the alleged invention, any long-felt but unsolved need in the art that was satisfied by the alleged invention, the failure of others to make the alleged invention, praise of the alleged invention by those having ordinary skill in the art, and copying of the alleged invention by others in the field. I understand that there must be a nexus—a connection—between any such secondary considerations and the alleged invention. I also understand that contemporaneous and independent invention by others is a secondary consideration tending to show obviousness.

28. I further understand that a claim is obvious if it unites old elements with no change to their respective functions, or alters prior art by mere substitution of one element for another known in the field, and that combination yields predictable results. While it may be helpful to identify a reason for this combination, common sense should guide and no rigid requirement of

finding a teaching, suggestion, or motivation to combine is required. When a product is available, design incentives and other market forces can prompt variations of it, either in the same field or different one. If a person having ordinary skill in the relevant art can implement a predictable variation, obviousness likely bars its patentability. For the same reason, if a technique has been used to improve one device and a person having ordinary skill in the art would recognize that it would improve similar devices in the same way, using the technique is obvious. I understand that a claim may be obvious if common sense directs one to combine multiple prior art references or add missing features to reproduce the alleged invention recited in the claims.

IV. BACKGROUND

A. Background of the Field Relevant to the '605 Patent

29. The '605 patent broadly relates to a method for operating a multiprocessor system including a plurality of data processing units (DPUs), and the optimization of power consumption using reconfigurable architectures and dynamic frequency control. In the following I will discuss only those aspects of the technology that I believe are directly relevant to the Challenged Claims of the '605 patent. The technology outlined below

represents a very small portion of the knowledge and understanding that one of ordinary skill in the art would have possessed.

1. Processor Clocking and Power Conservation

30. Central processing units (CPUs) or simply *processors* in computers contain an internal clock that guides or synchronizes its operation and ultimately constrains the speed of the processor. The clock is therefore an important component of a processor. The clock speed of a processor is often specified in, for example, GHz (gigahertz) or billions of clock cycles per second. The number of clock cycles per second is referred to as the clock frequency. Each processor instruction requires from one up to about a few tens of clock cycles to execute, or formally, to *retire*. The clock rate is also a significant factor in power consumption, in fact, power consumption is immediately proportional to the clock frequency.
31. Power consumption is especially an important factor in mobile computer systems. Most mobile computer or mobile device (e.g., phone) users want the longest possible battery life. The power consumption of a mobile device times its operating time equals the amount of battery charge used. Given that the battery charge capacity is largely fixed, given size constraints, operating time then is immediately determined by the power consumption of the mobile device. As a consequence, there has been considerable research and

development in how to optimize, and indeed minimize, power consumption in processors, with the goal of extending battery life as much as possible.

32. There are a number of ways one can optimize processors (or computer systems) for power conservation. One way is by monitoring and varying processor activity states or workloads. It was well known that microprocessors can have a wide range of activity states or workloads, ranging from not being used (inactive) to continuously processing instructions (a so-called high-power state). A computer system often has multiple processors and at any given time, some of the processors may be idle (inactive), some may be at a high-power state, and yet others at an intermediate power state.

33. Thus, in order to reduce power consumption in a processor, one can reduce the amount of time when a processor is in a high-power state.

34. This can be done in a variety of ways, such as by stopping or lowering the frequency of the clock of the microprocessor, halting operations (in other words, not executing instructions) within the microprocessor or at least reducing the queue of instructions (workload or simply, load) that must be executed, and lowering the voltage supply. The '605 patent explains that determining the "clock frequency for a microprocessor based on the state [was] known." Ex. 1003, 2:30–31.

35. Often times this is done when the system approaches its maximum tolerable power dissipation or a thermal operating limit.
36. This can be done at a system level, for example, by monitoring when the computer system is idle, and not being used, to lower the system's clock to a sleep clock level. A computer system is idle or approaching the idle state when the queue of instructions is empty or nearly so. In this case, the clock frequency can be reduced to a certain minimum necessary to enable the processor to retain its state and be able to respond when the workload increases again.
37. Alternatively, power conservation can be adapted for individual processing elements (or *processors*), rather than adjusting the system as a whole.
38. An individual processor is often made up of a variety of processing elements with different complexities, such as a multiplier and an adder.
39. A multiplier is an element that requires more intense computation than an adder. For example, an adder may require only a single clock cycle to execute whereas a multiplier may require four clock cycles to execute. Furthermore, a divider may require 32 clock cycles to execute.
40. It is inefficient to clock both the adder and multiplier functions using a high-speed clock if both results need to be completed at the same time.

41. It would be advantageous to clock the adder at a lower frequency than the multiplier in order to conserve power, without impacting system performance throughout.
42. Thus, in order to maximize power conservation, each processing element would ideally have its frequency determined by a measure of its inherent speed. That speed is impacted by process, voltage and temperature variations on an integrated circuit and/or printed circuit board (PCB) and hence clock estimation devices are used to ascertain an ideal clock frequency for a processing element.

2. Dynamic Optimization

43. Dynamic optimization of microprocessors is a component of performing power conservation. For example, a given processing element or processor's operating characteristics such as temperature, power consumption, and workload are constantly changing.
44. Thus, a system that can monitor these dynamic conditions, and adjust the system in response, will in turn optimize power conservation.
45. The power consumption of a device is directly related to the operating frequency and the voltage supplied to the device, as expressed in the equation: $P = CV^2f$, where P the power consumed, C is the total capacitance

of the components and wires that must be charged by the power supply, circuit, V is the power supply voltage, and f is the operating frequency.

46. Accordingly, a mobile system that has the ability to dynamically lower the operating frequency and/or the voltage supply to a processor will in turn lower the power consumption—extending the battery life.

47. Systems can monitor the workload a processor is to perform, such as the number of instructions or the complexity of instructions, in order to adjust the operating frequency. By monitoring the number of instructions in the instruction queue or buffer, a system can anticipate the degree of a workload faced by a processor and adjust the frequency accordingly.

48. For example, if one processor has a high backlog of instructions (workload) to be performed, it was known that it would be advantageous to increase the processor's clock frequency to a maximum.

49. Alternatively, if a second processor has little to no backlog of instructions, it was well known that it would be advantageous to clock that processor at a slower clock.

50. Operating temperature is a variable that is often monitored in processing systems, especially in computer systems. If the temperature gets too high, transistor switching speeds deteriorate and this may cause the system to fail to operate correctly at the designated clock frequency.

51. If a processor overheats, it can potentially cause irreparable harm to the processing chip. In particular, wires interconnecting the various transistors on the chip can melt and thus form unintended open circuits (breaks in the connectivity) that cause the chip to fail to operate correctly.
52. It was well known that as the operating frequency and voltage increase, so does the operating temperature since temperature relates directly to the power dissipation, which in turn depends on the voltage and frequency ($P = CV^2f$).
53. Thus, systems can monitor the operating temperature of a processor, and if it approaches a thermal threshold, the system can lower the clock frequency or reduce the power supply voltage to in turn reduce the operating temperature

3. Sequential Data Processing

54. There are many different types of processors.
55. Two types of processors with particular importance and relevance to the '605 patent are general-purpose processors (CPUs) and sequential data processors.
56. A general-purpose processor sequentially receives (loads) and executes (or retires) instructions. Each instruction is loaded and then retired before another instruction is loaded. Such a processor is termed a sequential instruction processor.

57. Sequential data processing on the other hand is done in such a way where the data that results from using arithmetic or logic operations (or a *processor*) is passed to another processor which starts execution only upon the receipt of the new data. Typically, a data flow graph is used, which indicates what other arithmetic or logic operations receive and operate on the data from a particular arithmetic or logic operation. In the usual case, a particular arithmetic or logic operation *fires* (or executes) whenever it has received a complete set of new data. Hence there is no preordained order in which particular arithmetic or logic operations are executed. This is in sharp contrast to a sequential instruction processor which executes instructions in a particular order.
58. In a sequential data processor, the final result is thus necessarily dependent on the intermediate results of the sequential processing elements.
59. In multiprocessor CPU systems, a typical mode of operation is parallel processing. Parallel processing distributes processing tasks between two or more processors or elements to concurrently process data.
60. Unlike sequential data processing, in parallel processing, the result from one processor or processing element is not passed directly to a subsequent processor.

61. In parallel processing, a given application to be executed can be divided into two or more independent tasks, in which each task is assigned to a distinct processor or processing element and the various processors or processing elements concurrently execute their given tasks.
62. This typically speeds up processing in proportion to the number of concurrently executing processors or processing elements

4. Reconfigurable Architecture

63. Reconfigurable computing architectures allow for greater flexibility and maximize utilization of limited resources. Reconfigurable architecture focuses on special purpose circuits with the ability to reconfigure the individual circuits as needed.
64. A given reconfigurable computing architecture is often made up of a number of basic building blocks, which can be programmed or configured for specific functions.
65. In order to optimize performance of computer systems, specialized co-processors have been coupled to more general-purpose processors or microprocessors.
66. Reconfigurable architectures utilize hardware, and when coupled to a more general-purpose microprocessor, can be adapted at runtime by the microprocessor for application-specific tasks.

67. One example of a reconfigurable architecture is a Field Programmable Gate Array (FPGA).

68. FPGAs are made up of a number of logic gates with a programmable interconnect network. The logic gates are typically look-up tables (LUTs) which can be programmed to implement any desired logic function of up to a certain number of inputs variables, e.g., 4-6 inputs.

69. Once configured by the microprocessor, the FPGA can run as a co-processor.

B. Summary of the '605 Patent

70. The '605 patent describes a method for operating a reconfigurable architecture multiprocessor system to allegedly reduce or optimize power consumption. '605 patent, 2:41–42.

71. The '605 patent focuses on a method for adjusting the clock frequency of the system or part of the system to optimize power consumption. *See id.*, [Claims 1–6]. The '605 patent explicitly notes that determining the clock frequency of a microprocessor based on the state is known from the area of mobile computers. *Id.*, 2:30–32. However, the '605 patent provides that it is beneficial to operate different cells using different clocking and to “achieve optimum data processing results, . . . it is suggested that clocking takes place depending on the state, which means that no clock is preselected jointly for

all cells based on a certain state, but rather an appropriate clock is assigned to each cell based on the state.” *Id.*, 2:53–54; 2:56–63.

72. The ’605 patent specification provides that in a preferred variant, the logic cells, or at least some of the cells, have at least one Arithmetic Logic Unit (ALU) and some cells contain at least one memory unit or register unit for data processing or configurations of the cell. *Id.*, 3:61–67. One skilled in the art would have understood that these specifications relate to reconfigurable architectures, such as FPGAs or the like because the ’605 specification makes it clear that not all of the logic cells have an ALU and memory unit, thus the cells are not individual processors. Further, using memory units to “configure” the cells is done in reconfigurable architectures.

73. According to the ’605 patent specification, different parameters may jointly establish the clock-determining state of the cells. *Id.*, 4:62–63. These parameters include the prioritization of tasks within a cell, power source, supply voltage, temperature measurements, and a user setting. *Id.*, 4:54–63.

74. The ’605 patent specification explains that such optimization can occur with: the controlled enabling and disabling of the clocks for the PAEs (processing array elements); applying different frequencies per PAE by using clock dividers, multipliers, or by oversampling; or with a configuration clock, in

which the operating frequency of PAEs can be made dependent based on temperature measurement, buffer filling levels, or battery charge state. *Id.*, 5:63–6:13; 6:14–67; 7:3–61.

C. Claims of the '605 Patent

Claim 1 is reproduced below:

[a] A method for operating a multiprocessor system comprising a plurality of data processing units,¹ each of the plurality of data processing units (a) including at least one Arithmetic Logic Unit (ALU) and a register unit and (b) being adapted for sequentially processing data, the method comprising:

[b] the multiprocessor system setting a clock frequency, of at least a part of the multiprocessor system to a minimum in accordance with a number of pending operations of a first processor;

[c] the multiprocessor system subsequently increasing the clock frequency of the at least the part of the multiprocessor system to a maximum in accordance with a number of pending operations of a second processor; and

[d] the multiprocessor system subsequently reducing the clock frequency of the at least the part of the multiprocessor system in accordance with (a) an

¹ On December 5, 2018 patentee filed a request for certificate of correction, granted on January 1, 2019, replacing “cells” with “units” in Claim 1, Column 15, Line 8. *See* '605 File History, 635 (Ex. 1004).

operating temperature threshold preventing over-temperature and (b) a hysteresis characteristic.

1.Claim 2

The method of claim 1 wherein: said clock frequency is determined in part by processor voltage.

2.Claim 3

The method of claim 1 wherein: said clock frequency is determined in part by available power.

3.Claim 4

The method of claim 1 wherein: said clock frequency is determined in part by a user setting.

4.Claim 5

The method of claim 1 wherein: said clock frequency is determined in part by available power and a user setting.

5.Claim 6

The method of claim 1 wherein: said number of pending operations is determined at by the fill level of one or more buffers.

D. Summary of the Prosecution History

75. The '605 patent issued as result of U.S. Application No. 13/653,639 (the "'639 application"). The '639 application was filed on October 17, 2012, claiming

priority, through multiple divisional and continuation patents, to a German Application DE 101 10 530 filed on March 5, 2001. Ex. 1003, Cover.

76. Claim 1 of the '639 application, which issued as the '605 patent, was rejected in a non-final office action on December 19, 2012, as being anticipated by the Williams reference, U.S. Patent No. 5,774,704. Ex. 1004 ('605 File History) at 239.

77. The applicant then responded to the office action on June 17, 2013, traversing the rejection under Williams, and argued that Williams failed to identically disclose “setting a clock frequency of at least part of the multiprocessor system to a minimum in accordance with a first processor load.” Ex. 1004 at 307. The applicant also argued that Williams is not considered with a processing load minimum, “[r]ather, under Williams, the flowchart is about reducing the processing load when it is at a maximum.” Ex. 1004 at 308.

78. The applicant expressly traversed this rejection by arguing that “It is from that maximum that the frequency is decreased – but the amount of decrease is not disclosed as being to a minimum.” Ex. 1004 at 308. Further, the applicant argued that Williams related to the cooling of the system based not on setting the clock frequency to a minimum, but by utilizing “cooling steps, not frequency adjustment steps.” Ex. 1004 at 308.

79. On September 20, 2013, Claim 1 was rejected in a subsequent office action under 35 U.S.C. § 101 as an abstract idea. Ex. 1004 at 318–23. The applicant responded, and submitted an amendment to the claims on October 23, 2013, amending claim 1 and adding 45 new claims Ex. 1004 at 324–34.

80. Again, on July 2, 2014, the patent office rejected claim 1 under 35 U.S.C. § 112(a) for failing to comply with the enablement requirement. Ex. 1004 at 357–60. On October 30, 2014, the applicant responded, amending claim 1, substituting “a number of pending operations” for the “load” limitation, and adding new claims, 47–64. Ex. 1004 at 365–82.

81. The applicant also noted that for enablement, the term “minimum” was disclosed in the specification where “. . . one or a plurality of cells are activated, if necessary, on a very low level, i.e., very slow clocking” Ex. 1004 at 378.

82. A notice of allowance issued on April 1, 2015. Ex. 1004 at 399.

V. LEVEL OF ORDINARY SKILL IN THE PERTINENT ART

83. I have been advised that there are multiple factors relevant to determining a person of ordinary skill in the art (“POSITA”), including the educational level of active workers in the field at the time of the alleged invention, the sophistication of the technology, the type of problems encountered in the art, and the prior art solutions to those problems.

84. As I explained above, the Challenged Claims of the '605 patent relate to optimization of power consumption using reconfigurable architecture, utilizing dynamic frequency control.

85. In my view a person of ordinary skill in the art would have been a person with at least a Master's degree in Electrical or Computer Engineering (or equivalent), and three years of experience with processor design.

86. As I explained above, the Challenged Claims of the '605 patent relate to optimization of power consumption using reconfigurable architectures.

VI. DETAILED INVALIDITY ANALYSIS

87. I have been asked to provide an opinion as to whether the Challenged Claims of the '605 patent are invalid in view of the prior art. The discussion below provides a detailed invalidity analysis of how the prior art references identified in Section I invalidate the Challenged Claims.

88. As part of my obviousness analysis, I have considered the scope and content of the prior art, and whether any differences between the alleged invention and the prior art are such that the subject matter, as a whole, would have been obvious to a person having ordinary skill in the art at the time of the alleged invention. I understand the claimed priority date of the alleged invention is March 5, 2001, and I have used that date as the time of the

alleged invention. I have also considered the level of ordinary skill in the pertinent art.

89. I describe in detail below the scope and content of the prior art, as well as any differences between the alleged invention and the prior art, on an element-by-element basis for the Challenged Claims.

VII. BACKGROUND ON PRIOR ART REFERENCES

90. Before providing a detailed analysis of how the prior art invalidates the challenged claims, I provide a brief summary of the asserted prior art.

A. Background on Nicol

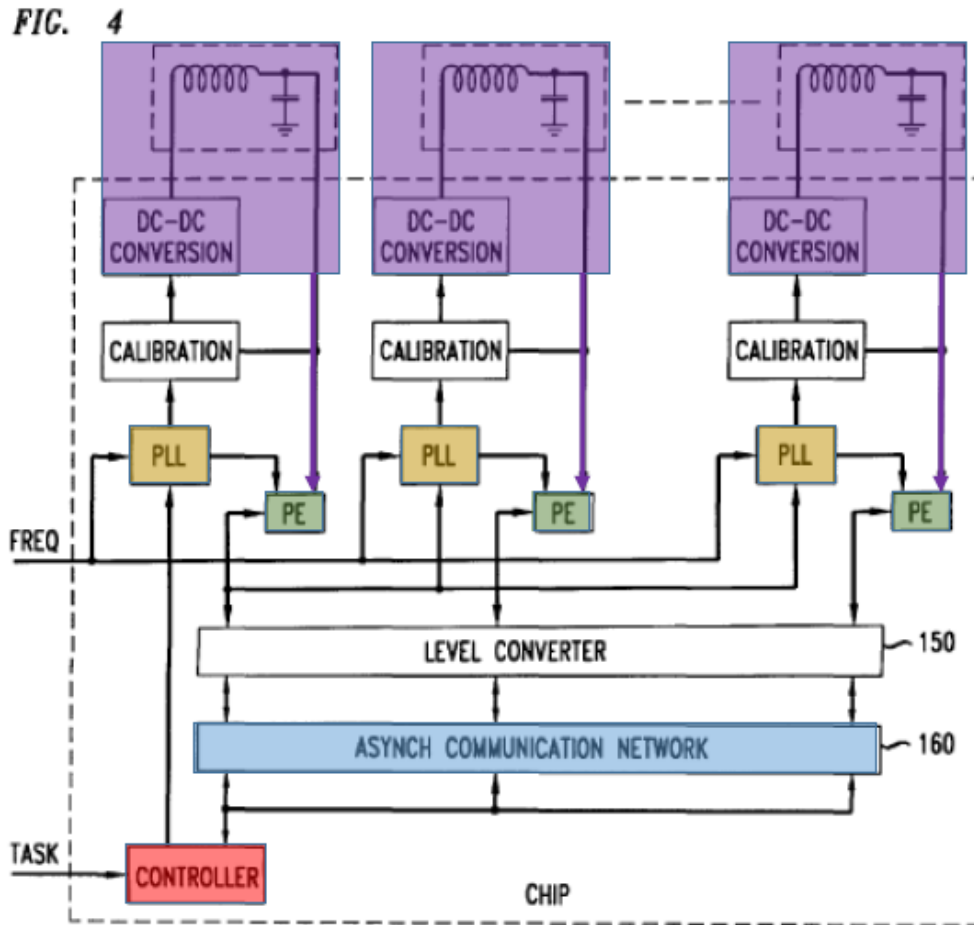
91. Nicol, titled “Power reduction in a multiprocessor digital signal processor based on processor load,” is directed to reducing power consumption of a multiprocessor system or chip through dynamic control of the processing load, clock frequency and supply voltage of the processing elements (PEs) of the system or chip. Ex. 1009, title, abstract.

92. Like the ’605 patent, Nicol explains that “[i]mproved performance of multiprocessor chips is achieved by dynamically controlling the *processing load* of chips [] so as to minimize overall power consumption. A controller in a multi-processor chip allocates tasks to the individual processors ..., lowers the clock frequency on the chip to as low as possible while ensuring proper operation, and finally reduces the supply voltage.” *Id.*, 2:18–27.

93. Further, Nicol teaches the ability to estimate processing loads based on the number of instructions of each of the tasks allocated to the PEs. *Id.*, 3:13–18 (“If the ***number of instructions*** that need to be executed for each task is known and made available to the operating system, a scheduler within the operating system can use this information to determine the best way to allocate the tasks to the available processors in order to balance the computation.”).

94. In Nicol’s multiprocessor system or chip, the PEs, are not restricted in their number or type. Further, the PEs can be configured to support “a mix of many digital signal applications.” *Id.*, 2:53–56.

95. Nicol’s annotated Figure 4 below is an exemplary embodiment of a multiprocessor chip having multiple PEs (green) configured to “exchange data” over a communication network (blue). *Id.*, 6:27–33. Each PE has its own individual power supply (purple) and clock generation circuitry (tan). This architecture allows the controller PE (red) to dynamically allocate tasks to selected PEs (green) and selectively program “the lowest operating frequency and corresponding lowest supply voltage for each PE.” *Id.*, 5:23–28, 5:66–6:2.

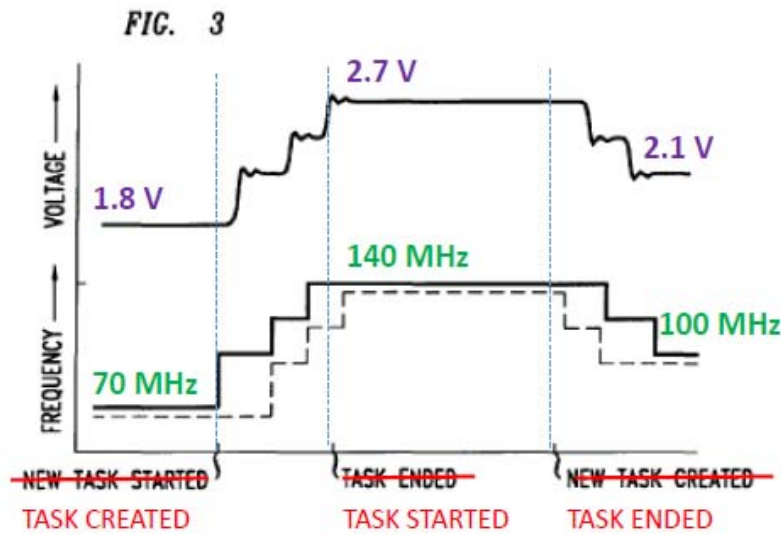


96.

97. Although Nicol's Figure 4 shows the PEs integrated in a single chip, Nicol explains that its power control scheme is "extensible to other system arrangements," including any system made of "a plurality of *separate* processor elements that operate at different frequencies and operating voltages, as well as components that are not typically thought of as processor elements." Ex. 1009, 6:34–39.

Further, like the '605 patent, Nicol's (corrected and annotated) Figure 3 illustrates the runtime clock frequency and supply voltage adjustments of the multiprocessor system as "to do" tasks are created, executed and then completed. Ex. 1009, 4:54–5:2. The sequence of task creation, task start and task end in the original Figure 3 was obviously wrong and has been corrected in red. Nicol explains in discussing Figure 3:

FIG. 3 demonstrates the timing associated with increasing Clk, Clk-L and V_{dd} -local ***when a new task is created*** [depicted as "task created" on corrected figure] ... Specifically, it shows the system operating at 70 MHz from a 1.8V supply when the load is increased in three steps to 140 MHz. When the ***2.7V supply*** is stable, as shown by the supply voltage plot, ***the new task is enabled for execution*** [depicted as "task started" on corrected figure]. Sometime thereafter according to FIG. 3, ***a task completes, which reduces the load on the multiprocessor*** [depicted as "task ended" on corrected figure]. The reduced load permits lowering the clock frequency to ***100 MHz*** and lowering the supply voltage to 2.1V. Ex. 1009, 4:58–66.



98.

99. This Figure show the multiprocessor system is initially in a low-activity or idle state (left), and operates at the low frequency (70 MHz) and low voltage (1.8V) to conserve power. *Id.* However, as new “to do” tasks appear, the controller determines the required completion time based on estimated processor load, and allocates the tasks to the PEs. Then, the controller raises the supply voltage of the PEs to a high voltage (2.7V), and also raises their clock frequency (to 140 MHz), when the PEs then transition to a high-performance state (middle), allowing them to execute the allocated tasks and complete them within the required completion time. *Id.* After completion of the tasks, the clock frequency and supply voltage of the PEs are slowly (in steps) reduced to minimize power consumption (right). *Id.*

100. Additionally, like the '605 patent, Nicol teaches that inactive PEs can also be selectively clock-gated and turned off to save power. *Id.*, 5:43–45. A POSITA would understand that gated clocking techniques, which are more often referred to as clock gating, means that the clock to a circuit block or processor is turned off. Since dynamic power consumption is proportional to the clock frequency, this reduces the dynamic power consumption to zero.

B. Background on Kling

101. The Kling reference describes a method and apparatus “for managing the power consumed in a computer system.” Ex. 1016 (Kling), 1:50–51. Kling teaches that computer systems typically include one or more processors and that modern computer systems are designed to operate at increasingly higher clock speed for higher performance. *Id.*, 1:16-21.

102. Further, Kling explains that, “as processor speed increases, the power consumed by the processor tends to increase as well.” *Id.*, 1:21-23. And increased power consumption could thus destroy devices due to over-temperature, and could significantly reduce battery life for mobile devices. Kling discusses conventional techniques to address those issues.

103. Kling explains that in particular, to address a critical over-temperature condition, a thermal sensor is typically used to monitor the temperature of a processor, as was well-known in the art. “If the processor temperature

exceeds a particular temperature, the processor is placed into a low power mode until it cools off. If these precautions are not taken, the processor may destroy itself by its own heat.” *Id.*, at 1:38-46.

104. Kling’s annotated Figure 1 illustrates a multiprocessor computer system that includes two processors 100 and 101 (green) and a power controller (red), as well as other integrated circuits (ICs) such as audio/video accelerators, external memory and chipsets. *Id.*, 2:65-3:19, Figure 1.

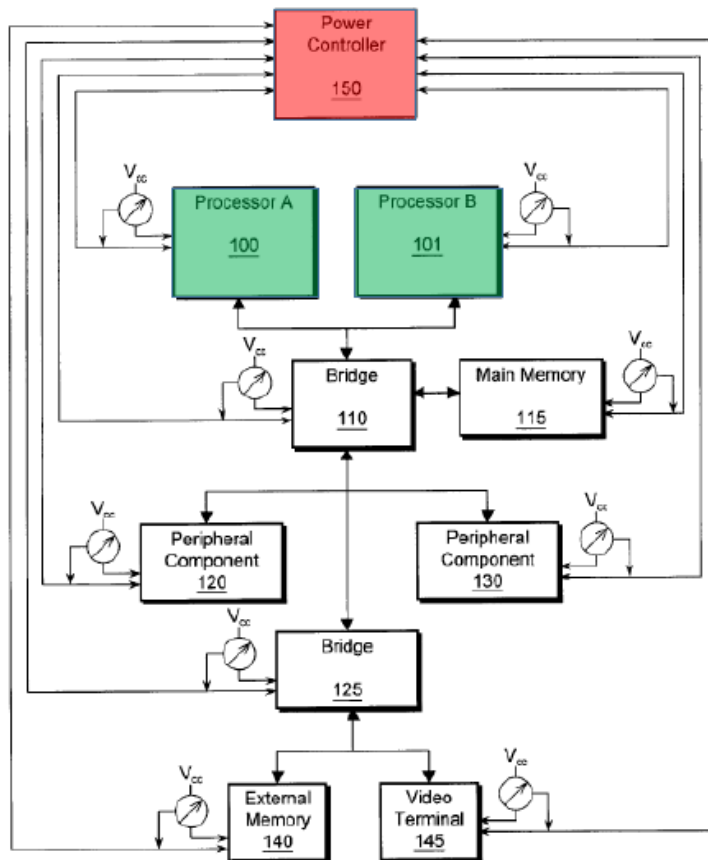


Figure 1

105.

106. Thus, the computer system of Kling’s Figure 1 “includes one or more power supplies, each of which provides power to one or more ICs.” *Id.*, 3:24-28. Kling explains that the system can “monitor the power consumed by a single IC or any number of ICs.” *Id.*, 3:28-29.
107. And if the measured power exceeds a threshold, Kling’s power controller (red) sends a “throttle signal” to “one or both of the processors 100 or 101” or other ICs of the computer system. *Id.*, 4:50-5:8. Kling explains that the ICs that receive the throttle signal must then reduce their power consumption. *Id.*, at 2:45-47. Notably, similar to the ’605 patent, a processor “*reduces its core frequency*” in response to receiving the throttle signal, as shown in the flow chart Figure 5. *Id.*, 2:47-51, Figure 5.

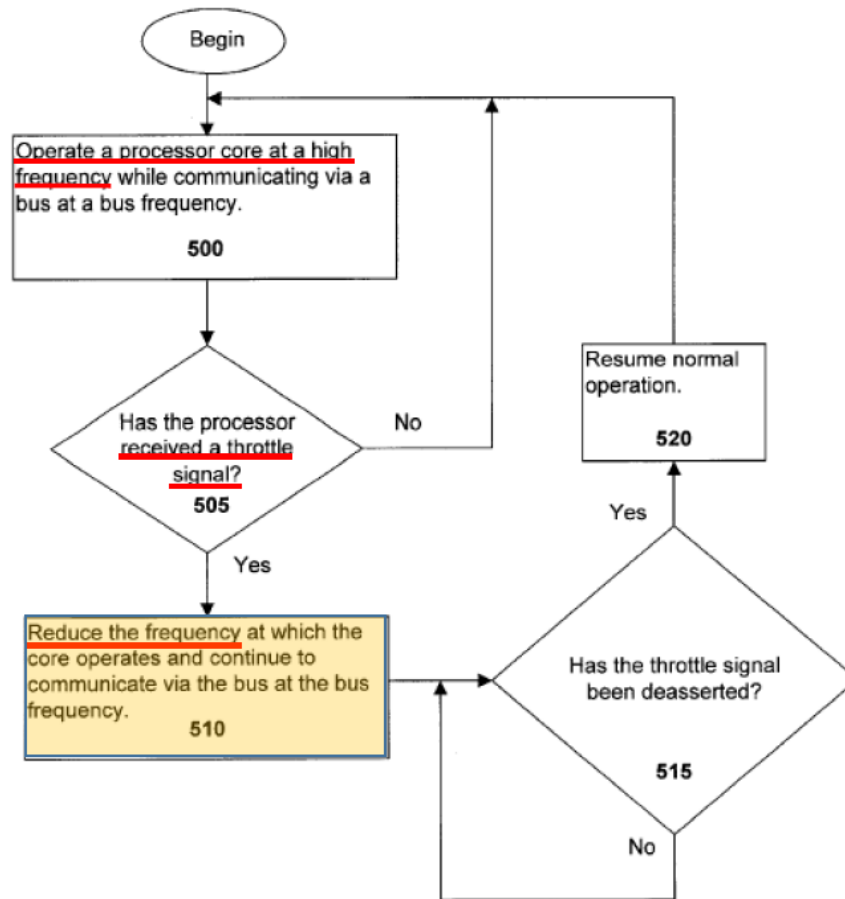


Figure 5

108.

C. Background on DeHon

109. The DeHon reference is directed to a reconfigurable microprocessor chip, called a DGPA-coupled microprocessor chip. Ex. 1012 (DeHon), Title. As illustrated in DeHon's annotated Figure 1, the chip "includes a *fixed logic processor* [core cell 10, red] as is common to perform logic operations." DeHon, 2:10–11. Similar to the '605 patent, "[m]inimally, the processor cell

10 includes an arithmetic logic unit (ALU), possibly a fixed processing unit (FPU), register files, instruction caches, and data caches.” *Id.*, 5:42–44

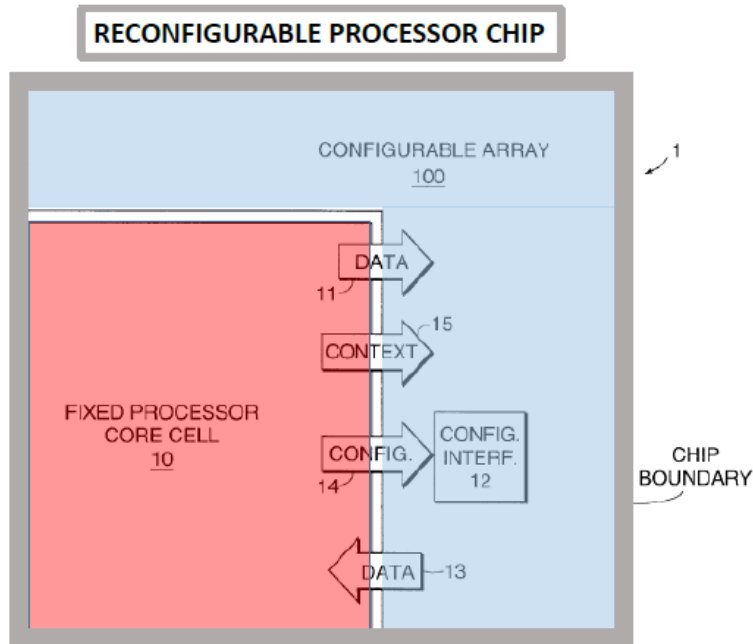


FIG. 1

110.

111. DeHon teaches that “[a] portion of the chip silicon real-estate, however, is dedicated to a *programmable gate array* [100, blue]. This feature enables *application-specific configurations*.” *Id.*, 2:12–15.

112. Further, similar to the ’605 patent, DeHon’s microprocessor chip is dynamically adjustable at runtime and “enables application-specific configurations” where the programmable gate array (blue) is “configurable to adapt to the particular time-changing demands of the microprocessor to provide the functionality required to best serve those demands.” *Id.*, 2:13–17.

113. Further, DeHon teaches that the configurable array initially receives configuration programming at power up or from the processor cell and, similar to the '605 patent, “[o]nce the proper configuration has been programmed into the array 100, the processor cell 10 feeds input data 11 to and receives the data output 13 from the configurable array 100.” *Id.*, 5:45–53. “For example, the operating system and applications specialize the configurable array to serve as a co-processor for compute-intensive functions such as compression, encryption, signal processing, or protocol processing.” *Id.*, 5:63–67.
114. Thus, different DGPA-coupled chips can thus be advantageously configured in different ways to handle a variety of compute-intensive functions “such as compression, encryption, signal processing, or protocol processing.” *Id.*, 5:63–67. Additionally, another advantage is that the configurable DGPA-coupled chips can replace or be deployed alongside traditional fixed processors. *Id.*, 7:20–25.

D. Background on Bhatia

115. The Bhatia reference is directed to the thermal management of a heterogeneous system, with processors of different types. Ex. 1011 (Bhatia), 2:31–36 (“Example components may include general or special-purpose processors ... application-specific integrated circuits (ASICs),

programmable gate arrays (PGAs), peripheral device controllers, and other types of devices.”).

116. Bhatia’s system (or parts of it) operates at different performance states, such as a high performance (HP) state and a low performance (LP) state, that are associated with different core clock frequency and voltage level settings. *Id.*, 2:41–48. In normal operation, the system may mostly operate in the HP state—at high voltage and frequency settings—for maximum performance.
117. However, when an over-temperature condition is detected, Bhatia teaches “throttling” back by cycling between the HP and LP states to reduce power consumption and lower system temperature below a target temperature. *Id.* at 2:51–55. Thus, Bhatia’s thermal management involves dynamic voltage and frequency control of processors. Bhatia’s Figure 6 illustrates the effectiveness of reducing temperature through dynamic voltage and frequency control of the processors. Ex. 1011, Figure 6.

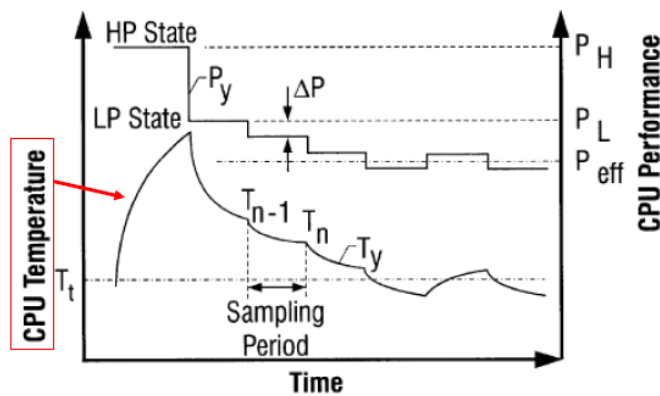


FIG. 6

118.

119. *Id.*, Figure 6 (annotated).

120. The Bhatia reference also discusses temperature sensing in detail, and discusses reducing the performance state “when an over-temperature condition is detected.” *Id.*, Abstract. Although Bhatia is primarily directed to thermal management of a system or chip, Bhatia also discusses a power management system. *Id.*, 12:26–37.

121. In fact, power and thermal management techniques are commonly implemented in computer systems. Many power and thermal management features are defined in the *Advanced Configuration and Power Interface* (ACPI) industry standard specification. Ex. 1011, 1:7–18 (citing the 1996 published version of ACPI). Specifically, ACPI specifies “*low activity state[s]*” (e.g., deep sleep, stop grant, C1, C2, or C3 state)” commonly used by power management systems to reduce power. *Id.*, 12:58–64. And in the C3 state for example, power is minimized because “no activity is performed by the processor except maintenance of the stored data in the processor’s internal cache.” Ex. 1011, 12:58–64.

VIII. CLAIMS 1-6 ARE OBVIOUS BASED ON NICOL IN VIEW OF KLING UNDER PATENT OWNER’S CONSTRUCTIONS

A. Reasons to Combine Nicol and Kling

122. In my opinion, a POSITA would have been motivated to combine the complementary teachings of Nicol and Kling.

123. As an initial matter, a POSITA would have found it obvious to combine the teachings of Nicol and Kling because both references are directed to interrelated thermal and power management aspects, which are critical in processor-based systems. Ex. 1009, Title (“Power Reduction in a Multiprocessor Digital Signal Processor Based on Processor Load”); Ex. 1010, 1:7–10 (“The present invention relates to computer systems and more particularly to limiting the power consumed in a computer system”); *see also* Ex. 1011 (Bhatia), 1:6–8 (“Different types of power and thermal management techniques have been implemented in system.”).
124. Additionally, Nicol and Kling describe the same flexible and distributed multiprocessor architecture, specifically the processors have their own power supplies, and software residing in a controller that is programmed to selectively control the power consumption of each processor, notably by dynamically adjusting the processor’s clock frequency during operation.
125. The Nicol reference teaches that the controller software is programmed to adjust both the clock frequency and supply voltage of the processors (PEs) based on their processing loads. Ex. 1009 (Nicol), 2:18-27. Similarly, Kling teaches a controller that is software programmed to adjust clock frequency of the processors, although based on sensor measurements that reflect

processing loads, such as power, voltage and current. Ex. 1010 (Kling) at 3:50–57.

126. However, the software differences are not material to the combination; in my opinion, a POSITA would have found it obvious to program Nicol’s controller to handle the additional software functionalities discussed in Kling.
127. Further, Nicol is focused on minimizing power consumption without discussing over-temperature protection. Because a POSITA would know that over-temperature protection is critical for high-performance systems, such as Nicol’s multiprocessor system and SoC, in my opinion a POSITA would have been motivated to consult Kling which describes conventional ways of providing over-temperature protection, as well as improvements.
128. One such improvement Kling teaches is by using simple power related measurements, such as measuring current, voltage, or power to detect excessive power consumption, i.e. when it is above a threshold, and use that information to “throttle” back processors (reducing the clock frequency) to reduce power consumption. Ex. 1010, 5:38–59.
129. In my opinion, a POSITA would have been motivated to implement this functionality in Nicol’s system for over-temperature protection and, more generally, to detect excessive power consumption and reduce it, which is

consistent with Nicol's objective of minimizing power consumption. Nicol, 2:18-22. Implementing this added functionality presents no risk, and does not require undue experimentation, has a good likelihood of success, and would yield predictable results because the added functionality is primarily add-on software to Nicol's existing controller software. *Id.*, 3:9-20

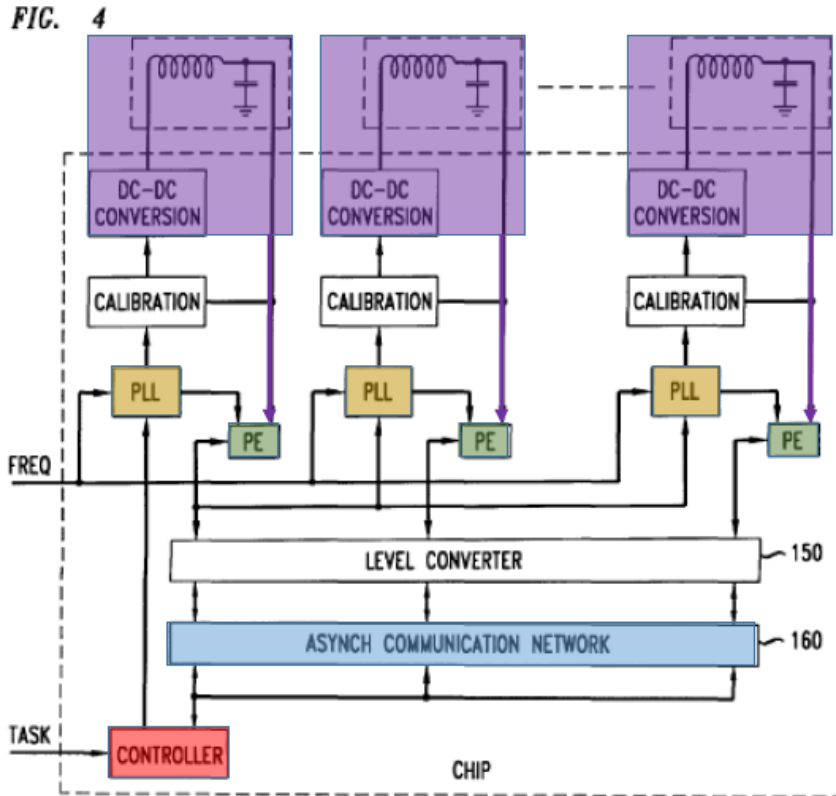
B. Claim 1

130. As I explain in detail below, in my opinion, Nicol, in view of Kling renders claim 1 obvious under Patent Owner's Claim Construction.

1. “[a] method for operating a multiprocessor system comprising”

131. In my opinion, Nicol discloses a method of operating a multiprocessor system. For example, the single-chip multiprocessor system illustrated in Nicol's Figure 4 embodiment (below, annotated) also includes a plurality of processing elements (PEs) (green). Ex. 1009, Figure 4.

132. Further, Nicol explains that each PE has its own power supply (purple) and clock generation circuitry (tan). Ex. 1009, 5:66-6:13. The PEs exchange data over a communication network (blue). *Id.*, 6:27–33 (explaining that the level converter 150 and asynchronous communication network 160 in Figure 4 resolve “the issue of synchronizing the exchange of data among the PEs of [the] multiprocessor chip.”).



133.

134. Nicol’s controller (red) “assigns the tasks given to the multi-processor chip of FIG. 4 among the PEs.” Ex. 1009, 6:11–13. Additionally, Nicol teaches that the operating system in the controller can “independently program the lowest operating frequency and corresponding lowest supply voltage for each PE.” *Id.*, 5:66-6:2.

135. Nicol explains that the operating system can also adjust the clock frequency and supply voltage to all PEs in unison. *Id.*, 6:67–7:3. This is more explicitly described in the context of the similar, but simpler Figure 2 embodiment, where the operating system “needs to ascertain the required

completion time [of the tasks], divide the collection of tasks as evenly as possible (in terms of needed processing time), consider the PE with the tasks that require the most time to carry out, and adjust the clock frequency [common to all PEs] to insure that the most heavily loaded PE carries out its assigned task within the required completion time.” *Id.*, 3:66–4:5.

136. Further, in my opinion, Kling also discloses a method of operating a multiprocessor system, such as the computer system illustrated in Kling Figure 1 (below) that includes a plurality of processors (green). Ex. 1010, Figure 1.
137. Kling teaches that the power consumption of the system is monitored and, if it exceeds a threshold, the system’s power controller (red) sends a “throttle signal” to command a reduction in power consumption. Ex. 1010, 3:28–29 and 4:50–5:8. Then, a processor “*reduces its core frequency*” in response to receiving the throttle signal. *Id.*, 2:47–51.

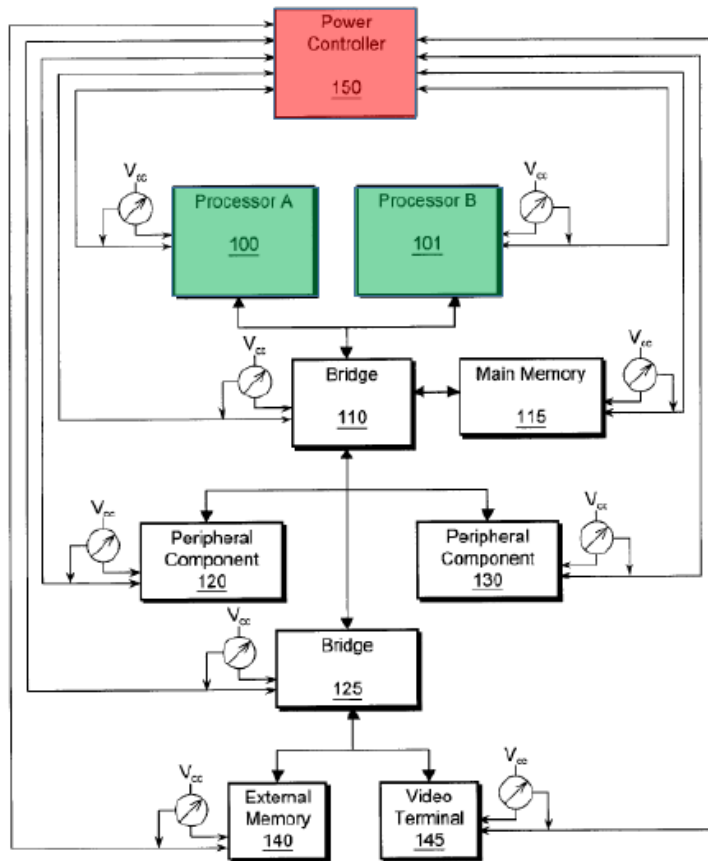


Figure 1

138.

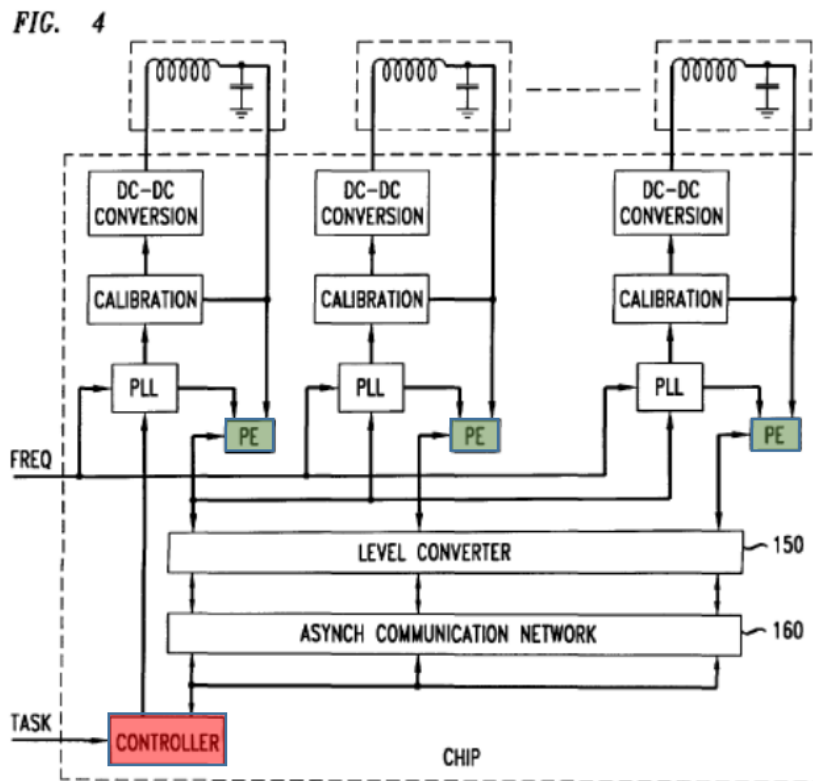
2. “a plurality of data processing units,”

139. In my opinion, Nicol discloses the method for operating a multiprocessor system comprising a plurality of data processing units.

140. For example, Nicol’s multiprocessor chip, illustrated in both Nicol’s annotated Figure 4 (below) and Figure 2, contains a plurality of processing elements (PEs) (green) that are data processing units (DPUs) because they each process data. Ex. 1009, Figure 4, 2:50–53 (explaining that the multiprocessor chip “contains *processing elements (PEs)* . . . and each PE

contains a central processing unit (CPU) and a local cache memory (not shown).”).

141. Nicol’s multiprocessor chip also includes a controller (red) that is a PE “that assigns the tasks given to the multi-processor chip among the PEs.” *Id.*, 3:10–20, Figure 4.



142.

143. Ex. 1009, Figure 4 (annotated).

3. “each of the plurality of data processing units (a) including at least one Arithmetic Logic Unit (ALU) and a register unit and”

144. In my opinion, Nicol renders obvious that each of the plurality of data processing units including at least one Arithmetic Logic Unit (ALU) and a register unit.
145. As previously discussed, Nicol discloses multi-processor chips containing processing elements (PEs), whereby “each PE contains a central processing unit (CPU) and a local cache memory (not shown).” Ex. 1009, 2:50–53. Thus, in my opinion, a POSITA would have understood that a PE having a CPU and cache, also includes arithmetic logic unit(s) (ALU) and registers. *See* Ex. 1022 (Hayes) at 33 and Figure 1.17.² Hayes Figure 1.17 (annotated

² I should note briefly that I am readily familiar with the Hayes textbook, titled: “Computer Architecture and Organization,” the third edition published in 1998 (Ex. 1022). The Hayes textbook is not new to me. Before the year 2000 I had consulted the Hayes textbook in connection with my studies. Notably, the publisher of Hayes—McGraw-Hill—is one of the largest publishers in the computer science and electrical engineering space and is highly respected. In addition to my previous knowledge with the Hayes textbook, I have further reviewed the text and consider it an authoritative source of information—as I previously did—as to what a POSITA would have known generally at the time

and reproduced below) shows a PE (labeled “data-processing unit DPU”) (green) with multiple ALUs (magenta) and associated registers (purple). *Id.*

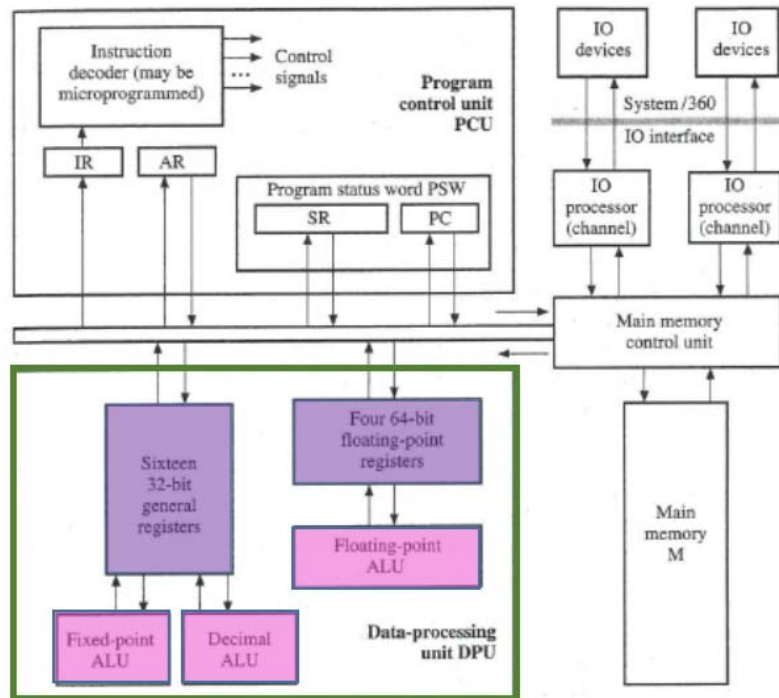


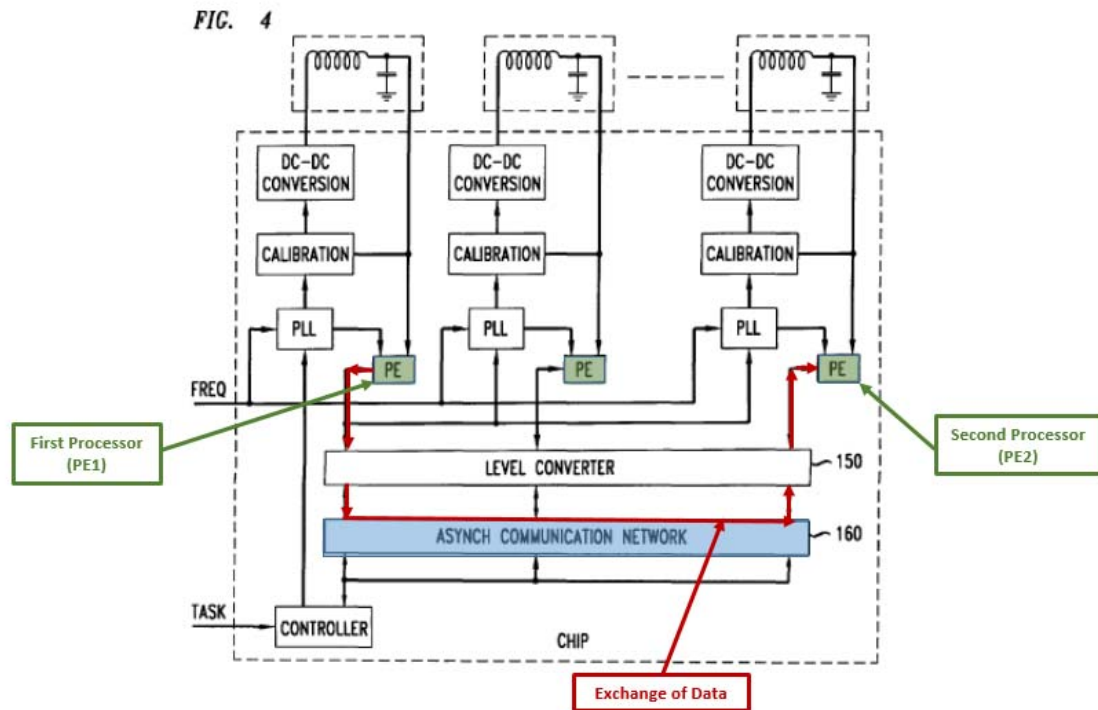
Figure 1.17
Structure of the IBM System/360.

146.

of its publication. Further, the labelled copyright date on the textbook is consistent with my understanding of when the Hayes textbook was publicly available. I have no reason to doubt the copyright date provided by McGraw-Hill.

4. “(b) being adapted for sequentially processing data.”

147. In my opinion, Nicol renders obvious this limitation under Patent Owner’s construction of the term “sequentially processing data” to mean “passing results onto one or more other data processing units which are subsequently processing data.” Ex. 1016 (JCC) at 11 and 17.
148. For example, Nicol teaches “*synchronizing the exchange of data* among the PEs [red and green] of a multiprocessor chip” through an “asynch[ronous] communication network” 160 (highlighted in blue in Nicol’s annotated Figure 4). Ex. 1009, 6:26–28, 6:14–18, The figure below shows the exchange of data (red) from PE1 through the asynchronous communication network (blue) to PE2 for further processing.



149.

150. In my opinion, asynchronous communication networks were well-known in the art. *See* Ex. 1022 (Hayes) at 491–93.

151. Thus, in my opinion, a POSITA would have understood that, in Nicol’s system, the data exchange between the PEs over the asynchronous communication network is performed in a clocked manner, as required by the claim. *Id.* (explaining that a communication is asynchronous “in the sense that the number of clock periods between [] events ... can be indeterminate, while the events themselves are *synchronized by the clock.*”). Nicol’s multiprocessor system is thus arranged with an asynchronous communication network (160 blue) *specifically* for the purpose of “passing

results onto one or more other data processing units which are subsequently processing data.” Ex. 1016 (JCC).

152. Such an asynchronous communication network would have been unnecessary—and Nicol’s multiprocessor system would have been of little practical use—if one PE could not pass its results to other PEs for further processing, as was commonly done in prior art multiprocessor systems. *See* Ex. 1022 at 159 and Figure 3.14 (first figure below) (describing a multiprocessor system in which a processor passes data to a floating-point co-processor for further processing), and at 290–291 and Figure 4.59 (second figure below) (describing a systolic array of PEs—one exemplary row of PEs is highlighted in green—in which “a flow of partial results” propagates from one PE to the next PE of a row to sequentially process data).

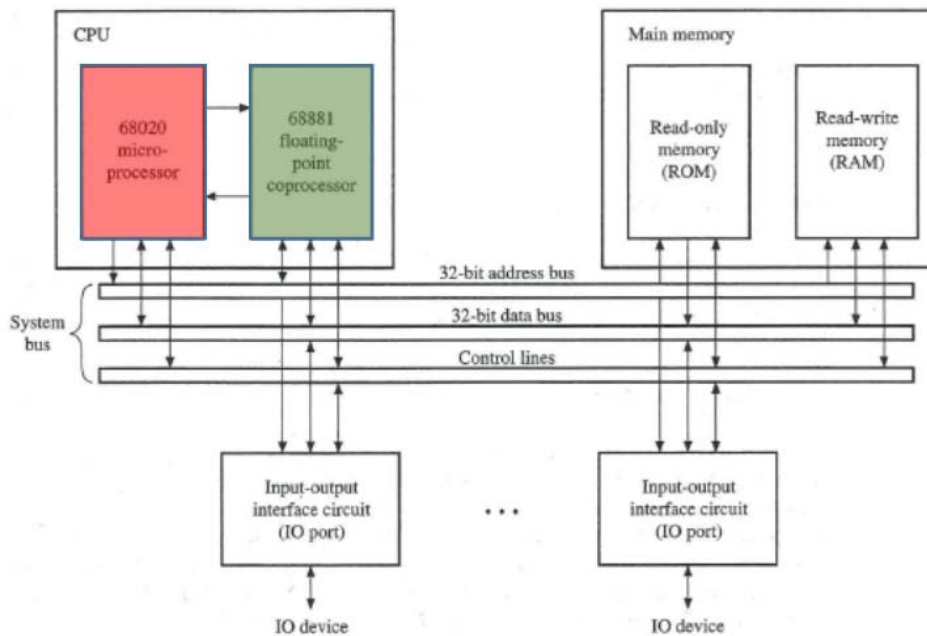
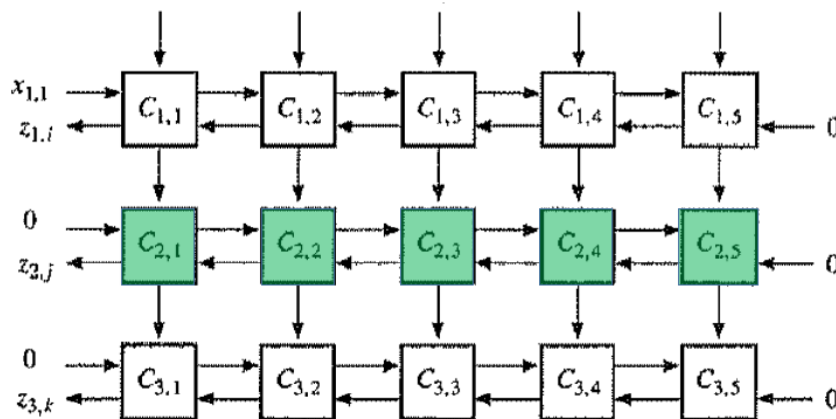


Figure 3.14
68020-based microcomputer with floating-point coprocessor.

153.



154.

155. Nicol's PEs communicate data, in part, because the system is intended to handle "a mix of many digital signal processing [DSP] applications," where DSP applications typically involve complex operations performed in a pipeline fashion, in which results from one PE are passed to another PE of the pipeline for further processing. Ex. 1009, 2:55–56.

156. One well-known example is in a graphics pipeline. In a graphics pipeline, graphical data (such as an image or pixels) is processed in successive stages through a graphics pipeline, illustrated in the figure below. Ex. 1026 (Williams), Figure 4. Williams' Figure 4 illustrates data generated by the "Pixel Ops" processor are sent to the "Rasterization" processor for further processing, then to the "Per Fragment Ops" processor for additionally processing, and finally to the "Frame Buffer." Ex. 1026 (Williams), 10:56–67.

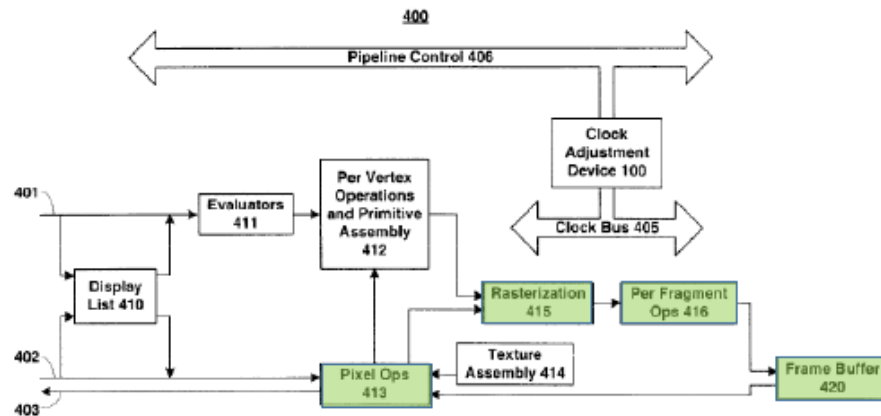


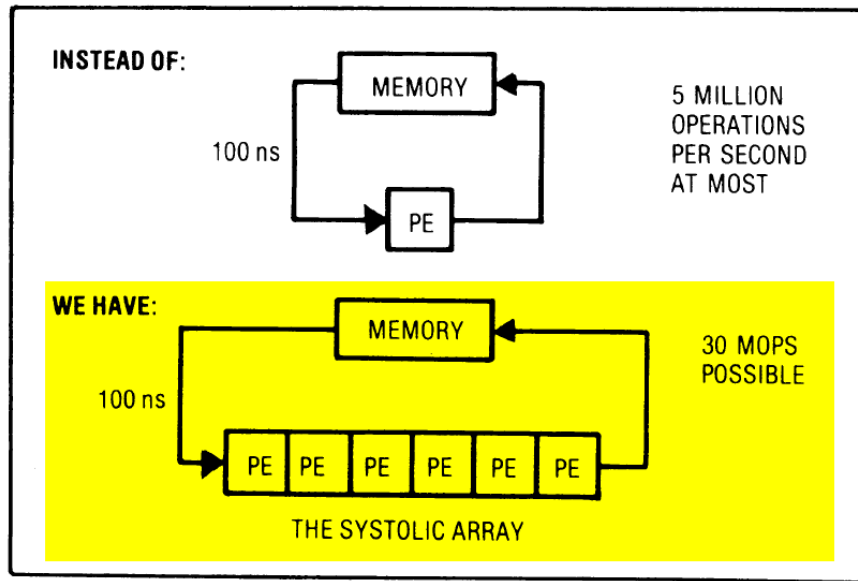
Fig. 4

157.

158. In addition, a POSITA would have known that the PEs in Nicol's multiprocessor system can be configured to sequentially process data, such as in the well-known systolic architecture. Ex. 1014 (Kung, H.T., *Why Systolic Architectures?*, Computer Vol. 15 (Pub. January 1982)) ("Kung"), 45 (describing "numerous systolic designs [] known" in the 1980s), 37 ("In a

systolic system, data flows from the computer memory in a rhythmic fashion, *passing through many processing elements before it returns to memory*, much as blood circulates to and from the heart.”).

159. For example, Kung’s Figure 1 below illustrates PEs that are arranged in a systolic array and pass data from one PE to the next, with each PE performing a separate operation on the received data. Ex. 1014, Figure 1; *see also* Ex. 1011 (Hayes textbook), at 290 (“Closely related conceptually to arithmetic pipelines are the data-processing circuits called *systolic arrays* [] formed by interconnecting a set of identical data-processing cells in a uniform manner. Data words flow synchronously from cell to cell, with each cell performing a small step in the overall operation ... Systolic processors have been designed to implement various complex arithmetic operations ...”).

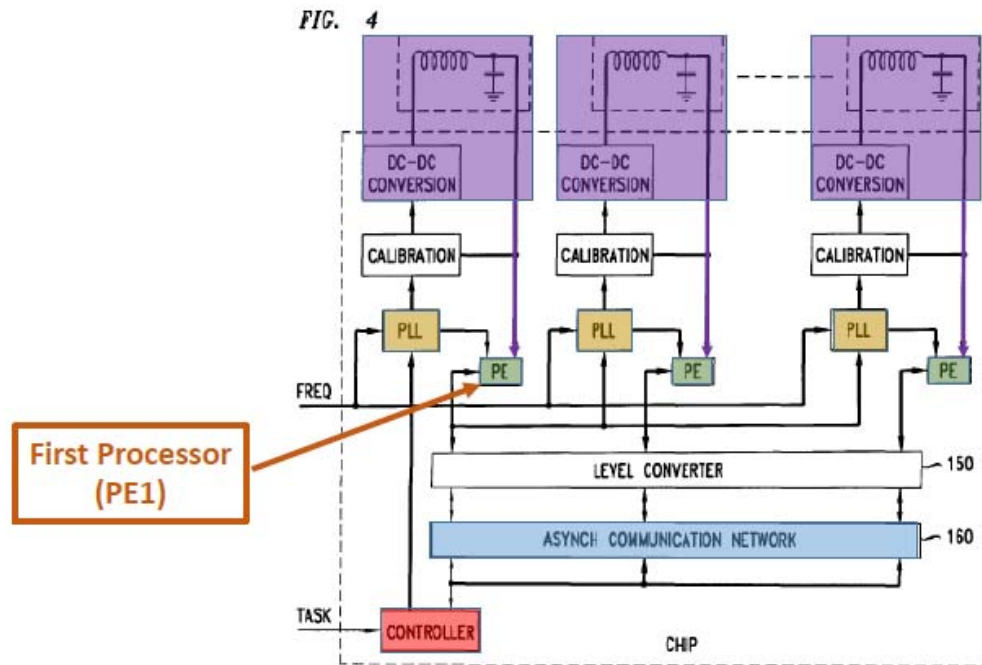


160. **Figure 1. Basic principle of a systolic system.**

5. “the multiprocessor system setting a clock frequency, of at least a part of the multiprocessor system to a minimum in accordance with a number of pending operations of a first processor;”

161. In my opinion, Nicol discloses the multiprocessor system setting a clock frequency, of at least a part of the multiprocessor system to a minimum in accordance with a number of pending operations of a first processor.

162. In looking at Nicol’s annotated Figure 4 below, Nicol explains that the controller (red) in its multiprocessor system sets a clock frequency to a minimum according to a number of pending operations (instructions) of each of the PEs of the system, including the number of pending operations of PE1 (“first processor”).



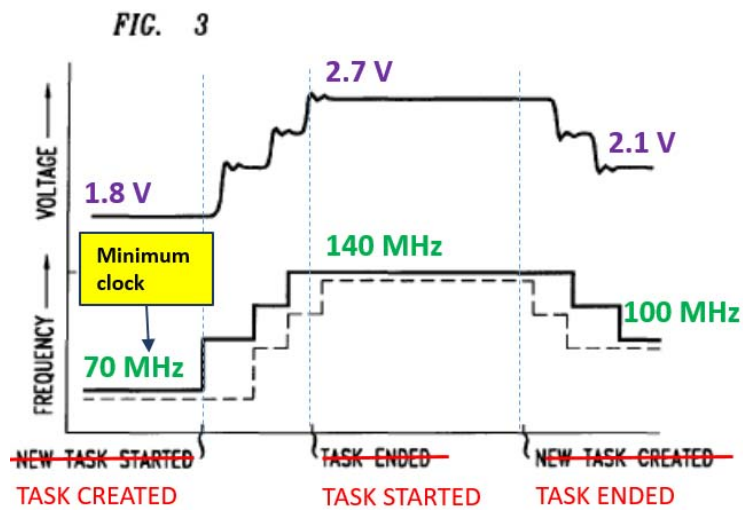
163.

164. In fact, Nicol teaches that the controller “allocates tasks to the individual processors [PEs, including PE1] so that it can equalize the processing load among the [PEs, including PE1], and then the controller *lowers the clock frequency* on the chip to *as low a level as possible* while assuring proper operation, and finally reduces the supply voltage.” Ex. 1009, 2:23–27.

165. Nicol further teaches that the controller takes into account the number of pending operations (instructions) of each PE—including of PE1 (“first processor”)—in its determination of processing load and task allocation. Ex. 1009, 3:13–18 (“If *the number of instructions* that need to be executed for each task is known and made available to the operating system, a scheduler within the operating system can use this information to determine the best

way to allocate the tasks to the available processors in order to balance the computation.”).³

166. Nicol’s annotated Figure 3 below illustrates setting the clock frequency of the multi-processor system to a minimum (70 MHz in Figure 3) when the PEs, including PE1, have a light load with few (or no) pending operations, and a new “to do” task is not yet assigned to the PEs for execution. Ex. 1009, 3:31–33, 4:54–62, 5:23–28.



- 167.
168. Although Nicol’s Figure 3 above illustrates a minimum clock frequency of 70 MHz, Nicol further teaches setting the clock frequency to zero if none of

³ Patent Owner also represented to the patent office during prosecution of the ‘605 patent that “a number of operations pending [] is equivalent to the processor load.” Ex. 1004 at 377.

the PEs are needed because all tasks have been completed and there are no new tasks or pending operations for any PE. Ex. 1009, 5:42–44 (“the operating system employs gated clocking techniques to ‘shut down’ PEs that are not needed.”).

169. Like the gating techniques described in Nicol, in my opinion, a POSITA would have known that Nicol’s power management system with dynamic frequency control, would set the clock frequency of processing elements that are not needed (or had a low number of pending operations) to a “minimum.”
170. For example, the well-known industry standard, “the Advanced Configuration and Power Interface (ACPI) Specification” was commonly used and referred to by a POSITA. Ex. 1020 (ACPI Spec.); *see also* Ex. 1011 (Bhatia), 1:6–17 (explaining “the Advanced Configuration and Power Interface (ACPI) Specification, Rev. 1.0., published on Dec. 22, 1996, that provides an interface between the operating system of a system and hardware devices to implement power and thermal management.”).
171. Thus, such a “minimum” clock frequency teaching is discussed in the ACPI Specification, and it was well-known in the art to set the clock frequency to a minimum in a low power mode. Ex. 1011, 12:58–65 (“the C3 state defined under the ACPI specification, [whereby] the external clock BCLK to the

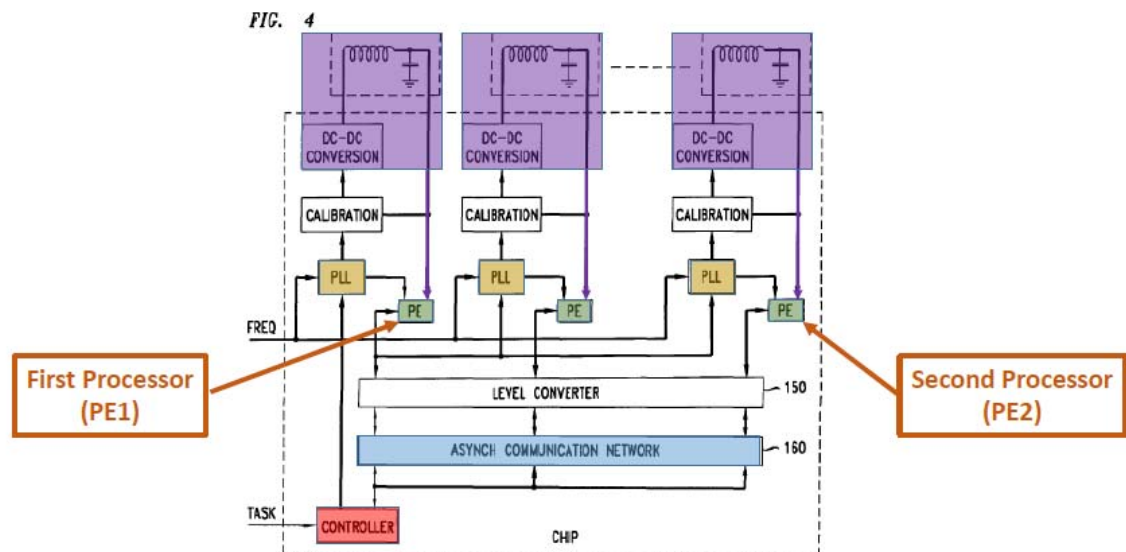
processor is disabled so that no activity is performed by the processor *except maintenance of the stored data in the processor's internal cache.*").

172. In sum, it is my opinion that a POSITA would have understood that Nicol's system could have referenced the teachings of the ACPI, and implemented a similar low power mode in Nicol's system, thus setting the clock rate of inactive PEs to a "minimum" clock rate. In my opinion, a POSITA would have been motivated to implement a low power mode such as ACPI's C3 state in any computer system, such as Nicol's system, because such a mode preserves memory contents, which serves to speed up the computer system's transitioning back to full operation upon "waking up" from a low power mode.

6. "the multiprocessor system subsequently increasing the clock frequency of the at least the part of the multiprocessor system to a maximum in accordance with a number of pending operations of a second processor;"

173. In my opinion, Nicol discloses the multiprocessor system subsequently increasing the clock frequency of the at least the part of the multiprocessor system to a maximum in accordance with a number of pending operations of a second processor.
174. As previously discussed, the controller in Nicol's multiprocessor system adjusts the clock frequency of the system according to a number of pending operations of each of the PEs—including the number of pending operations

of PE2 (“second processor”)—identified in Nicol’s annotated Figure 4 below. *Id.*; Ex. 1009, 2:23–27 (discussing allocating tasks to the PEs and setting an appropriate clock frequency for task execution) and 3:13–18 (discussing using “*the number of instructions* that need to be executed for each task” for task allocation).

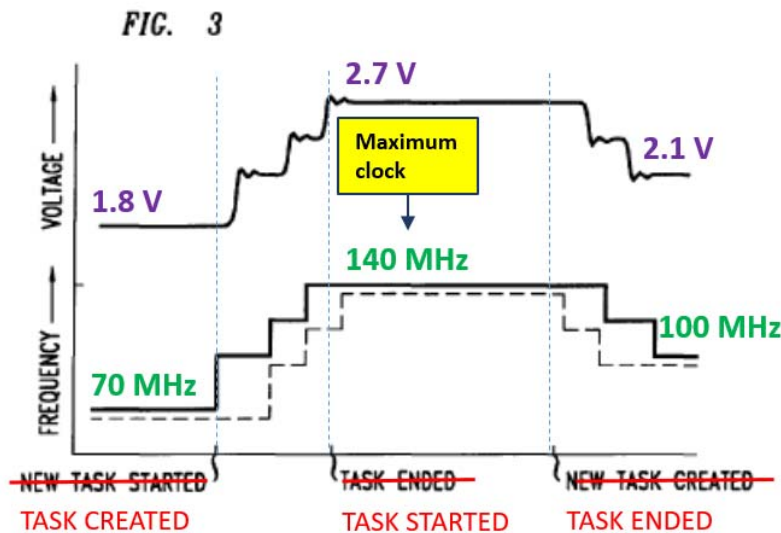


175.

176. Specifically, the controller takes into account the number of pending instructions for each PE—including PE1 (“first processor”) and PE2 (“second processor”)—because it sets the clock frequency based on identifying the processing requirement of “the most heavily loaded PE.” Ex. 1009, 3:66–4:5 (explaining that the operating system “needs to ascertain the required completion time [of the tasks], divide the collection of tasks as evenly as possible (in terms of needed processing time), consider the PE with the tasks that require the most time to carry out, and *adjust the clock*

frequency [common to all PEs] to insure that *the most heavily loaded PE* carries out its assigned task within the required completion time.”).

177. Additionally, Nicol’s annotated Figure 3 below illustrates the multiprocessor system increasing the clock frequency of the PEs of the multiprocessor system from 70 MHz (minimum clock frequency) to 140 MHz (maximum clock frequency) prior to the execution of a high load, *i.e.*, involving a high number of pending operations for the PEs. Ex. 1009, 3:31–33, 4:54–62, 5:23–28.
178. Nicol explicitly explains that each chip has a “*maximum [frequency]* that the chip can support” (Ex. 1009, 3:63–64), as would have been well-known to a POSITA, and that “[f]or a maximally utilized system, all of the available processors [PEs] *ought to be operating at full speed* [maximum frequency] when satisfying the maximum load encountered by the system.” *Id.*, 2:65–3:4.



179.

7. “and the multiprocessor system subsequently reducing the clock frequency of the at least the part of the multiprocessor system in accordance with (a) an operating temperature threshold preventing over-temperature and (b) a hysteresis characteristic.”

180. In my opinion, Nicol in view of Kling renders obvious the multiprocessor system subsequently reducing the clock frequency of the at least the part of the multiprocessor system in accordance with (a) an operating temperature threshold preventing over-temperature and (b) a hysteresis characteristic.

181. The Kling reference explains that processing systems must be able to rapidly detect and mitigate over-temperature situations. Ex. 1010, 1:38–46 (“To address the thermal issue [in modern computer systems] . . . processor packages include a thermal sensor to monitor the temperature of the processor. *If the processor temperature exceeds a particular threshold*, the

- processor is placed into low power mode until it cools off. If these precautions are not taken, the processor may destroy itself by its own heat.”).
182. Thus, as I previously explained in Section VIII.A (Reasons to Combine Nicol and Kling), in my opinion, a POSITA would have found it obvious, and necessary, to include over-temperature control in the multiprocessor system of Nicol.
183. Notably, in my opinion, a POSITA would have found it obvious to detect over-temperature in Nicol’s multiprocessor system by comparing the system’s measured temperatures against a temperature threshold, as required by the claim.
184. For example, the Kling reference explains that this was in fact conventionally done in prior art systems. Ex. 1010, 1:43–45 (“If the processor temperature *exceeds a particular threshold*, the processor is placed into low power mode until it cools off.”); *see also* Ex. 1011, 1:6–12 (same).
185. Further, in my opinion, a POSITA would have been motivated to use this conventional technique because Nicol’s multiprocessor system already includes on-chip temperature sensor(s) for “tracking temperature.” Ex.

1009, 4:11–13 (“the operating system [in the controller] can reduce the supply voltage even further by *tracking temperature* ...”).⁴

186. Although Nicol discusses using the tracked temperatures for on-chip voltage scaling, a POSITA would have known that the same temperatures can obviously be used to also detect over-temperature conditions.

187. Additionally, as the Kling reference explains, and as would have been well-understood by a POSITA, once over-temperature is detected, the overheated chip or system needs to be “placed into low power mode until it cools off.” Ex. 1010, 1:44-45.

188. Thus, a POSITA would have known that placing Nicol’s multiprocessor system in a low power mode requires *reducing the clock frequency* of its PEs to reduce power consumption, as also taught by Nicol and Kling and as would have been well-known to a POSITA. Ex. 1009, 1:26–28 (“The power consumption of a CMOS circuit increases linearly with operating frequency

⁴ See also Ex. 1009, 1:41–43 (“Voltage scaling as a function of temperature has been incorporated into the Intel Pentium product family as a technique to achieve high performance at varying operating temperatures and process corners. ... The approach uses an *on-chip temperature sensor* ...”).

- and quadratically with supply voltage.”); Ex. 1010, 2:43–52 (explaining that when a processor receives a “throttle signal” requesting it to reduce its power consumption, the processor “reduces its core frequency”).
189. Next, the Kling reference also discusses the use of hysteresis to improve robustness and “reduce the occurrence of power oscillation,” as would also have been well-known in the art. Ex. 1010, 5:50–58.
190. For example, a POSITA would have known that hysteresis is commonly implemented by comparing the same signal against two thresholds, as shown in Kling’s annotated Figure 2A (below). Ex. 1010, Figure 2A, 5:51–54 (“In accordance with the present invention, the lower threshold may be set to a predetermined value *to provide hysteresis* to reduce the occurrence of power oscillation between the upper and lower thresholds.”); Ex. 1021 (Kennedy, M. & Chua, L., *Hysteresis in Electronic Circuits: A Circuit Theorist’s Perspective*, International Journal of Circuit Theory and Applications, Vol. 19, (1991)) at 471 (explaining that circuits with hysteresis characteristics were “indispensable” and commonly used “in noisy environments and to guard against false triggers”).

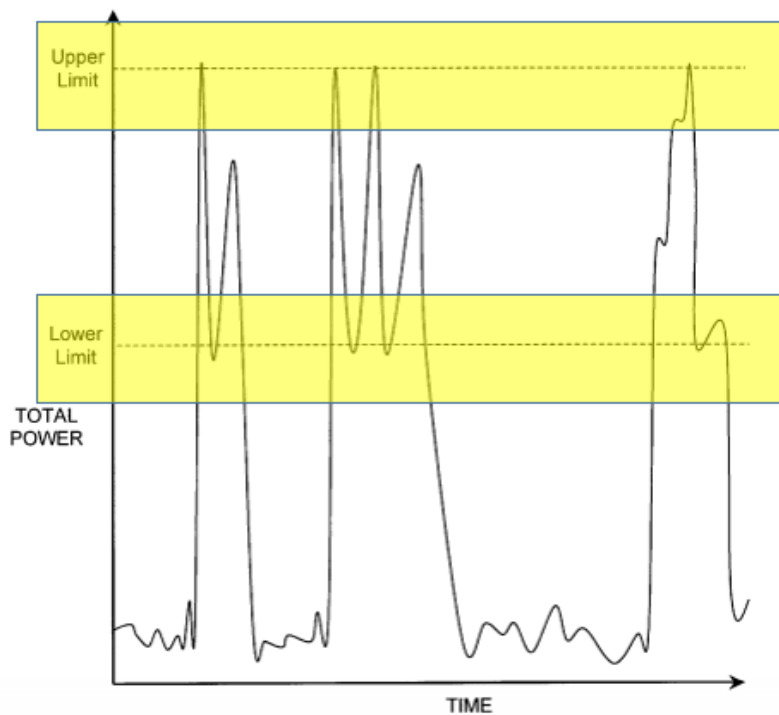


Figure 2A

191.

192. Thus, in my opinion, a POSITA would have found it obvious to apply the well-known two-threshold method (providing hysteresis) from Kling in implementing an over-temperature monitoring system in Nicol for the same well-known reason—to reduce unwanted oscillations.

C. Dependent Claim 2

1. “The method of claim 1 wherein: said clock frequency is determined in part by processor voltage.”

193. In my opinion, Nicol in view of Kling renders obvious the method of claim 1 wherein: said clock frequency is determined in part by processor voltage.

See supra VIII.B.

194. For example, as previously discussed, in my opinion, a POSITA would have found it obvious to add Kling’s power reduction based on measured power (“PRBMP”) functionality to Nicol’s system. *See supra* Sections VIII.A and VIII.B.
195. For example, Kling discloses an implementation of PRBMP whereby the “a parameter that is approximately proportional to the consumed power, *such as voltage* . . .” is tracked. Ex. 1010, 2:31–37. Further, current consumed by a processor can also be tracked by measuring “a voltage across a resistor” of the processor as would have been, in my opinion, well-known in the art. Ex. 1010, 3:57–59.
196. Thus, Kling teaches that if the voltage exceeds a threshold—indicating excessive power consumption—the controller sends a “throttle signal” to the processor to reduce its clock frequency, which reduces power consumption. *Id.*, 2:48-51 (explaining that “a processor ... reduces its core frequency” upon receiving a throttle signal sent by the controller).
197. In my opinion, Kling thus discloses reducing a clock frequency based in part on a voltage in a processor (“processor voltage”), and a POSITA would have understood that PRBMP can also be easily implemented in Nicol’s system with no risk.

D. Dependent Claim 3

1. “The method of claim 1 wherein: said clock frequency is determined in part by available power.”

198. In my opinion, Nicol in view of Kling discloses the method of claim 1 wherein: said clock frequency is determined in part by available power.
199. As discussed above in Section VIII.A and VIII.C, in my opinion, a POSITA would have found it obvious to add Kling’s power control functionality to Nicol’s system.
200. Specifically, Kling teaches measuring the power (*e.g.*, current or voltage across an appropriate resistor) of a processor and then comparing the measured power against an “upper limit” threshold, as shown in Kling’s annotated Figure 2A below. Ex. 1010, Figure 2A.

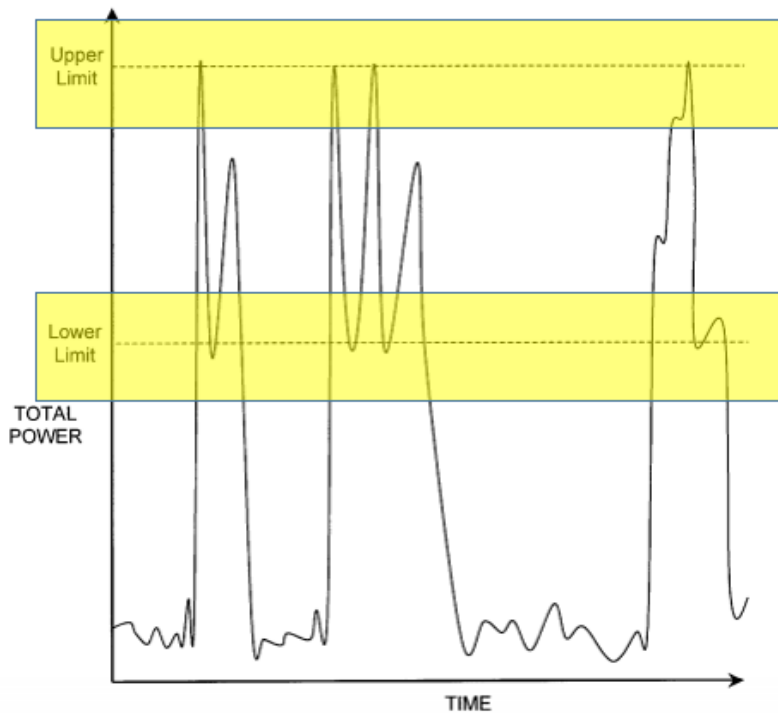


Figure 2A

201.

202. Kling explains that the upper limit in Figure 2A is the available power to the processor. Ex. 1010, 5:21-25 (explaining that “*the upper limit* is a constant value ... associated with an *approximately maximum power that can be reliably consumed by the computer system ...*”).

203. Then, Kling explains if the measured power reaches the upper limit, the controller sends a “throttle signal” to the processor to instruct it to reduce its clock frequency to reduce power consumption. Ex. 1010, 5:38–40, 2:48-51.

204. In my opinion, a POSITA would have understood that this functionality can be readily implemented in Nicol’s system with a reasonable expectation of success.

205. This additional functionality taught by Kling involves relatively simple add-on software to Nicol's more complex task scheduler software residing in the controller, and relies on well-known measurement techniques, such as measuring a voltage across an appropriately placed resistor.
206. In addition, in my opinion, a POSITA would have been motivated to add the functionality, involving a relatively simple add-on, and relying upon well-known techniques, such as measuring voltage to a processor, or voltage drop across a resistor, and apply it to Nicol's system as it protects against over-temperature and can be generally used to control power consumption, consistent with Nicol's objective, which would yield predictable results.

E. Dependent Claim 4

1. "The method of claim 1 wherein: said clock frequency is determined in part by a user setting."

207. In my opinion, Nicol in view of Kling renders obvious the method of claim 1 wherein: said clock frequency is determined in part by a user setting.
208. As discussed above in Section VIII.A, VIII.C and VIII.D, in my opinion, a POSITA would have found it obvious to add Kling's power control functionality to Nicol's system.
209. As previously discussed, Kling teaches measuring the power (*e.g.*, current or voltage across an appropriate resistor) of a processor and comparing the measured power against an "upper limit" threshold as well as a "lower limit"

threshold, as shown in Kling's annotated Figure 2A below. Ex. 1010, Figure 2A.

210. In fact, these thresholds are user settings. Ex. 1010, 5:57–59 (“Alternatively, *the lower threshold may be modifiable by a user* or automatically adjusted within the computer system to, for example, reduce power oscillation.”).

211. Kling explains that if the measured power reaches the upper limit, the controller sends a “throttle signal” to the processor to instruct it to reduce its clock frequency to reduce power consumption. Ex. 1010, 5:38-40, 2:48-51.

212. Further, “[o]nce [the] lower threshold is reached, the throttle signal may be deasserted, and, in response, the IC resumes normal operation.” *Id.*, 5:48-50.

213. Thus, in this system, the clock frequency is determined in part by user settings, *i.e.*, the upper and lower threshold power limits.

F. Dependent Claim 5

1. “The method of claim 1 wherein: said clock frequency is determined in part by available power and a user setting”

214. In my opinion, Nicol in combination with Kling renders obvious the method of claim 1 wherein: said clock frequency is determined in part by available power and a user setting.

215. First, as previously discussed, Nicol in combination with Kling discloses a system wherein the clock frequency is determined in part by available power. *See supra* Section VIII.D.
216. Second, as previously discussed, Nicol in combination with Kling discloses a system wherein the clock frequency is determined in part by a user setting. *See supra* Section VIII.E.

G. Dependent Claim 6

1. “The method of claim 1 wherein: said number of pending operations is determined at by the fill level of one or more buffers.”

217. In my opinion, Nicol renders obvious the method of claim 1 wherein: said number of pending operations is determined at by the fill level of one or more buffers.
218. For example, Nicol teaches estimating processing loads based on the number of instructions for each of the “to do” tasks to be allocated to the PEs. Ex. 1009, 3:13–18 (“If the *number of instructions* that need to be executed for each task is known and made available to the operating system, a scheduler within the operating system can use this information to determine the best way to allocate the tasks to the available processors in order to balance the computation.”).

219. Thus, in my opinion, a POSITA would have understood that the instructions that need to be executed are stored in an instruction queue or buffer, and that the fill level of the buffer corresponds to the number of instructions. Ex. 1022 (Hayes) at 155 and Figure 3.11 (showing an “instruction queue”).

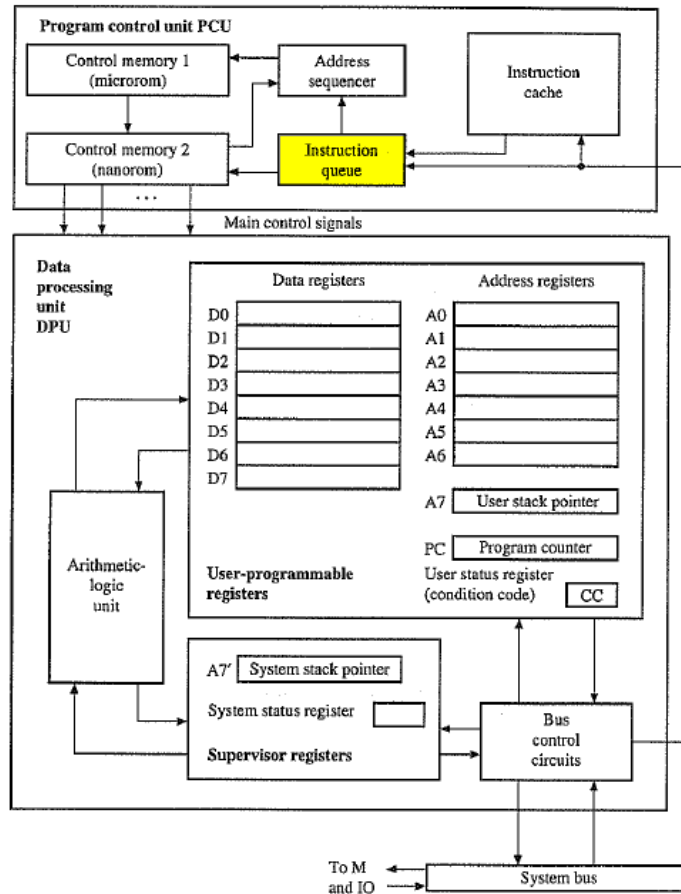


Figure 3.11
Organization of the 68020.

220.

221. Further, the applicant stated during the prosecution of the '605 patent that *“one of ordinary skill in the art would understand that the fill level of a data buffer [is] representative of the number of waiting operations or processor load.”* Ex. 1004 at 377.

222. Thus, in my opinion, a POSITA would also have known that Nicol detects the number of pending operations of the PEs by inspecting the fill level of buffers because it was well known in the art that the number of pending operations could be determined by monitoring queues, such as the fill level of buffers that feed work to the processors. Ex. 1018 (Cline), 4:57–60 (“Usage can be monitored in various ways, such as by taking an average of recent workload and assuming the average represents the near-term workload, *or by monitoring queues that feed work to the device.*”); Ex. 1017 (Hong), 1704 (explaining dynamic supply voltage can be adjusted based on “processing load as measured by *the number of data samples queued in the input (or output) buffer.*”).

223. Thus, in my opinion, a POSITA would have known that the number of pending operations or load of the PEs in Nicol are determined by the fill level of a buffer.

IX. CLAIMS 1 – 6 ARE OBVIOUS BASED ON NICOL IN COMBINATION WITH KLING, AND DEHON UNDER PETITIONER’S CONSTRUCTIONS

224. As I explained above, I understand that the parties dispute the construction of the terms “data processing unit” and/or “data processing unit . . . adapted for sequentially processing data” and “to a minimum.”

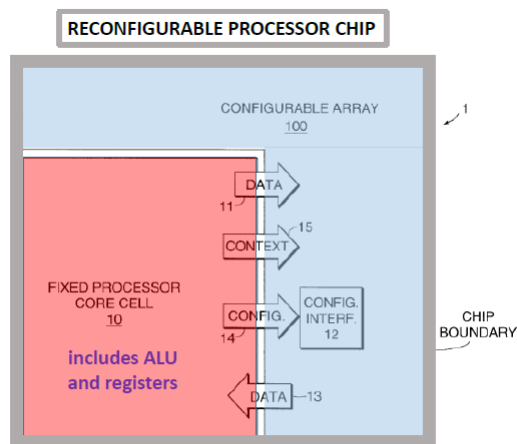
225. I am informed that Petitioner proposes the following construction for “data processing unit” and/or “data processing unit . . . adapted for sequentially processing data”: “reconfigurable and sequential data processors where the data results from one processor are fed to another, for each processor to perform a separate computation.” I am also informed that Petitioner proposed the following construction for “to a minimum”: “to a level no more than is required for the preservation of memory contents or the like.”
226. In my opinion, the rationale from Section VIII, which is my analysis under Patent Owner’s Construction, applies to Section IX, which is my analysis under Petitioner’s Constructions. In Section IX, Petitioner adds the DeHon reference to satisfy Petitioner’s claim construction requiring that the data processing unit be reconfigurable. DeHon teaches a reconfigurable data processing chip, as detailed under VII.C (Background on DeHon).
227. I am aware that Petitioner’s proposed constructions only affects two claim limitations in challenged claim 1. Sections IX.A and IX.B below explains that Nicol in view of Kling and DeHon renders obvious these two claim limitations.
228. For all other claim limitations and claims, Nicol in view of Kling and DeHon satisfies the claim elements the same way as Nicol in view of Kling

satisfies them as set forth under Section VIII, just that the Nicol's PEs are replaced by DeHon's reconfigurable microprocessor.

A. Reasons to Combine Nicol, Kling, and DeHon

229. As previously discussed, in my opinion, a POSITA would have been motivated to combine Nicol and Kling. *See supra* Section VIII.A.
230. Further, it is my opinion that a POSITA would have been motivated to combine the system of Nicol and Kling with DeHon.
231. As an initial point, a POSITA would have found it obvious to combine the Nicol and Kling system with DeHon because Nicol itself provides a motivation to combine its teachings with DeHon. Nicol explains that its multiprocessor system is intended for use in a variety of digital signal processing applications. Ex. 1009, 2:53–56 (“A real-time operating system resides in PE 100 and allocates tasks to the other PEs *from a mix of many digital signal processing applications.*”).
232. Thus, in my opinion, a POSITA would have been motivated to consult and apply DeHon, because DeHon teaches an “inventive microprocessor [that] can serve as the principle *building block for a wide range of applications* including personal computers, embedded systems, application-specific computers, and general and special-purpose multiprocessors” Ex. 1012, 2:18–24.

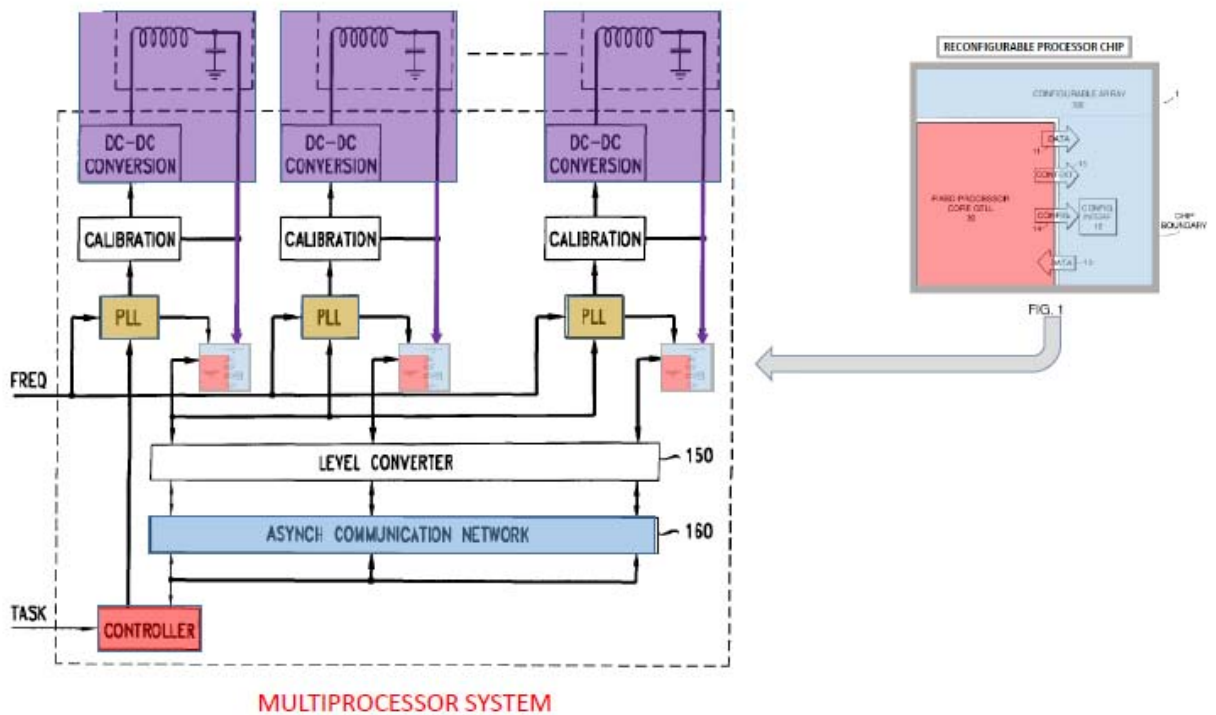
233. DeHon explains that such an arrangement “yields application acceleration and in system-specific functions.” Ex. 1012, Abstract. Thus, because Nicol explains that its system is adapted for use in a variety of digital signal processing applications, in my opinion, a POSITA would have been motivated to apply the teachings of DeHon, which discloses a flexible, reconfigurable, microprocessor chip, and use such reconfigurable microprocessors as PEs in Nicol’s system. DeHon’s annotated Figure 1 below illustrates DeHon’s reconfigurable processor chip.



234.

235. Although Nicol’s preferred embodiment integrates all PEs on a single multiprocessor chip, Nicol expressly teaches using its dynamic power management scheme in a multiprocessor system having discrete processor chips. Ex. 1009, 6:34–39 (“The principles disclosed above for a multiprocessor is extendible to other system arrangements. This includes systems with a plurality of *separate* processor elements ...”).

236. Thus, in my opinion, a POSITA would have understood the flexibility of Nicol's system and that it encompasses the use of reconfigurable technology. In my opinion, it would have been obvious to a POSITA that the PEs disclosed in Nicol's system can be reconfigurable.
237. And because, in my opinion, a POSITA would have known that Nicol's PEs could be reconfigurable, it would have thus been obvious to a POSITA to replace each of Nicol's PEs with DeHon's reconfigurable microprocessor chip, illustrated in DeHon's Figure 1. Ex. 1012, FIG. 1.
238. The annotated figure below combines Nicol's annotated Figure 4 with DeHon's annotated Figure 1 to illustrate Nicol's multiprocessor system in which the PEs are DeHon's reconfigurable processor chips.



239.

240. In my opinion, a POSITA would have been motivated to substitute fixed PEs with DeHon's reconfigurable PEs because the resulting reconfigurable multiprocessor system has significantly improved capacity to handle any "mix of many digital signal processing applications" through rapid reconfiguration provided by DeHon's microprocessors. Ex. 1009, 2:55–56; Ex. 1012, Abstract ("[Reconfiguration] enables application-specific configurations to allow adaptation to the particular time-changing demands of the microprocessor and provide the functionality required to best serve those demands.").

241. This simple substitution would have had a reasonable likelihood of success, because Nicol teaches a flexible power management scheme that is agnostic to the types of PEs—whether fixed or configurable, homogeneous or heterogeneous. Ex. 1009, 6:34–35 (“The principles disclosed above for a multiprocessor is *extendable* to other system arrangements.”).
242. Further, in my opinion, a POSITA would have found it obvious to combine Nicol and DeHon because DeHon itself provides a motivation to combine its teachings with Nicol. For example, DeHon explains that a portion of its reconfigurable chip “is dedicated to a *programmable gate array*. This feature enables *application-specific configurations*.” Ex. 1012, 2:12–14, which, in my opinion, a POSITA would have connected with Nicol’s “mix of many digital signal processing applications.” Ex. 1009, 2:53–56.
243. DeHon’s reconfigurable chips can thus be advantageously configured in different ways to handle a variety of compute-intensive functions “such as compression, encryption, signal processing, or protocol processing.” Ex. 1012, 5:63–67. Another advantage is that the configurable DGPA-coupled chips can replace or be deployed alongside traditional fixed processors. *Id.*, 7:20–25. Thus, in my opinion, a POSITA would have been motivated to use DeHon’s reconfigurable chips in Nicol’s multiprocessor system because

DeHon's reconfigurable chips are designed to replace or be deployed alongside traditional fixed processors, which minimizes integration risks.

244. Additionally, in my opinion, a POSITA would have been motivated to combine Nicol and DeHon because both relate to the same field of invention, optimizing processor systems by dynamically controlling processors. Ex. 1009, Abstract, [Claim 45]; Ex. 1012, 5:61–67. Specifically, DeHon relates to dynamically reconfiguring the programmable array, depending on the particular application to be performed. Ex. 1012, 5:61–63 (“Each application program configures the reconfigurable array in a manner most beneficial to that application.”). Similar to DeHon, Nicol's controller dynamically allocates tasks to the individual PEs based on the application to be performed, like the configuration of the array in DeHon, and dynamically programs the respective clock frequencies and supply voltages for each PE. Ex. 1009, Abstract.

245. Thus, for all of the reasons stated, in my opinion, a POSITA would have been motivated and found it obvious to combine the Nicol and Kling system with DeHon.

B. Claim 1

246. The two claim limitations with respect to claim 1, with claim terms in dispute are discussed below. The remainder of the claim limitations are rendered obvious based on the same reasoning from Section VIII.

1. “a plurality of data processing units . . . being adapted for sequentially processing data”

247. As previously discussed, Nicol discloses data processing units being adapted for sequentially processing data under Patent Owner’s Construction. *See supra* Section VIII.B.4. Petitioner’s construction incorporated all of the above, and adds DeHon in combination with Nicol and Kling, such that the combined system discloses “reconfigurable and sequential data processors where the data results from one processor are fed to another, for each processor to perform a separate computation.”

248. As an initial matter, the combination of Nicol, Kling, and DeHon discloses reconfigurable processors. For example, in DeHon’s annotated Figure 1 below, DeHon depicts a microprocessor with a fixed core 10 (red) connected to a reconfigurable array 100 (blue). Ex. 1012, Figure 1; *see also id.*, 2:25–27 (“In specific implementations, the programmable gate array is configurable by the processor rather than some off-chip configurer.”); [Claim 19].

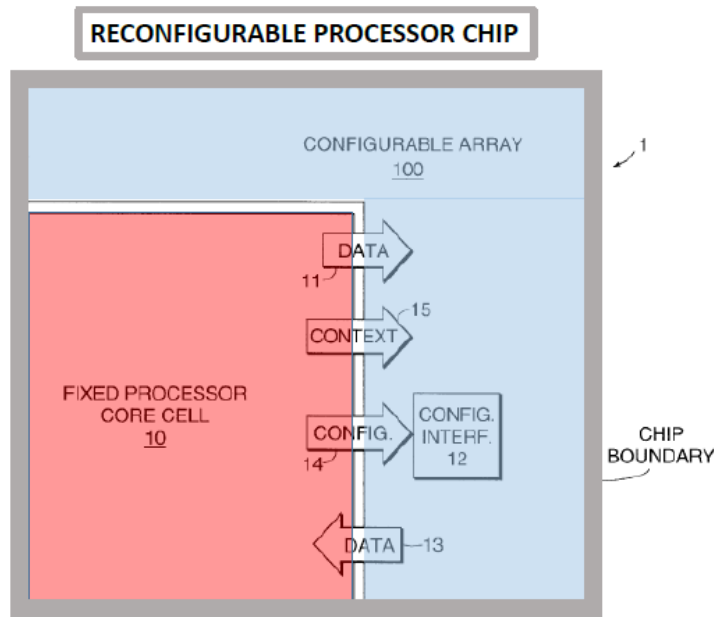


FIG. 1

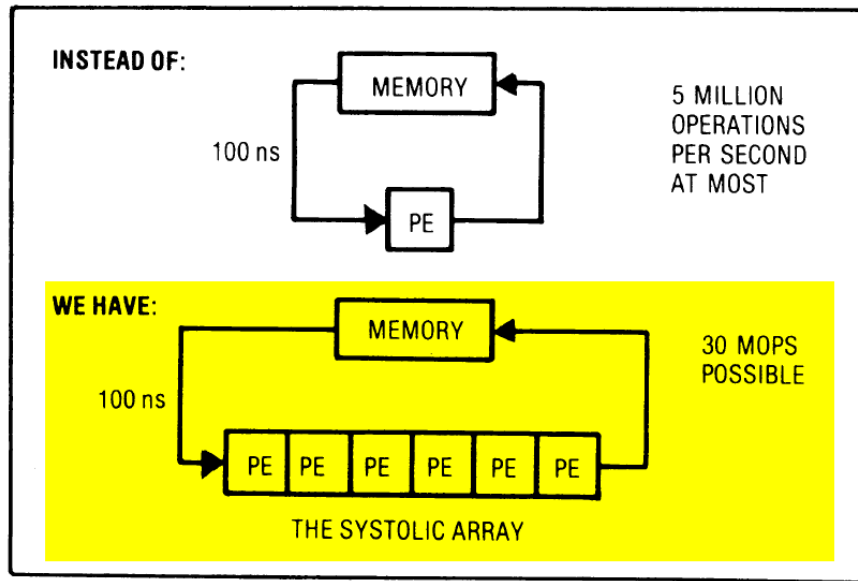
249.

250. Thus, in my opinion, the combination of Nicol, Kling, and DeHon discloses reconfigurable processors.

251. Further, in my opinion, the combination of Nicol, Kling, and DeHon discloses the reconfigurable processors are adapted for sequentially processing data where the results from one processor are fed to another, for each to perform a separate computation.

252. It is clear that Nicol teaches that the PEs in its multiprocessor system—which, in my opinion are readily replaced with are DeHon’s reconfigurable PEs, and in this combination—are interconnected through a communication network over which they “exchange data” to perform “a mix of many digital signal processing systems.” Ex. 1009, 2:55–56, 6:26-33, Figure 4.

253. This type of data exchange allows data to flow from one PE (the source PE) to another PE (destination PE) for further processing, as was common in digital signal processing systems. In my opinion, a POSITA would have known that the DeHon reconfigurable PEs can be configured to sequentially process data in a manner similar to the well-known systolic architecture. *See* Ex. 1014 (Kung, H.T., *Why Systolic Architectures?*, Computer Vol. 15 (Pub. January 1982)) (“Kung”), 45 (describing “numerous systolic designs [] known” in the 1980s), 37 (“In a systolic system, data flows from the computer memory in a rhythmic fashion, *passing through many processing elements before it returns to memory*, much as blood circulates to and from the heart.”).
254. Kung’s Figure 1 below illustrates PEs that are arranged in a systolic array and pass data from one PE to the next, with each PE performing a separate operation on the received data.



255. **Figure 1. Basic principle of a systolic system.**

256. .

2. “the multiprocessor system setting a clock frequency, of at least a part of the multiprocessor system *to a minimum* in accordance with a number of pending operations of a first processor;”

257. In my opinion, as previously discussed, Nicol discloses this claim element under Patent Owner’s Construction. *See supra* Section VIII.B.5.

258. Further, in my opinion, Nicol renders obvious this claim element under Petitioner’s construction of “to a minimum” to mean “to a level no more than is required for the preservation of memory contents or the like.”

259. For example, Nicol teaches minimizing power consumption of a chip, whereby the controller PE can “independently program the lowest operating frequency and corresponding lowest supply voltage for each PE.” Ex. 1009, 5:66–6:2.

260. Additionally, Nicol further discloses that its operating system “employs gated clocking techniques to ‘shut down’ PEs that are not needed.” *Id.*, 5:42–45.
261. Consistent with Nicol, in my opinion, it was well-known in the art to use the standards set forth in “the Advanced Configuration and Power Interface (ACPI) Specification.” Ex. 1020 (ACPI Spec.); *see also* Ex. 1011 (Bhatia), 1:6–17 (explaining “the Advanced Configuration and Power Interface (ACPI) Specification, Rev. 1.0., published on Dec. 22, 1996, that provides an interface between the operating system of a system and hardware devices to implement power and thermal management.”).
262. The ACPI specification also teaches a low power mode, defined as the C3 state. Ex. 1011, 12:58–65 (“the C3 state defined under the ACPI specification, [whereby] the external clock BCLK to the processor is disabled so that no activity is performed by the processor *except maintenance of the stored data in the processor’s internal cache.*”).
263. It is my opinion that the ACPI was (and still is) an industry standard specification, and would have been well-known to a POSITA.
264. Thus, in my opinion, a POSITA would have understood that Nicol would have conformed to ACPI and would have implemented ACPI’s C3 state and thus setting the clock rate of inactive PEs (or DeHon’s microprocessor chips

replacing Nicol's PEs) to a minimum clock rate necessary to preserve memory contents, *e.g.*, cache memory contents.

265. In my opinion, a POSITA would have been motivated to implement a low power mode such as ACPI's C3 state in any computer system, including Nicol's system, because such a mode preserves memory contents, which serves to speed up the computer system's transitioning back to full operation upon "waking up" from a low power mode. And the C3 state is akin to a "light" sleep state.

X. CLAIMS 1–6 ARE OBVIOUS IN VIEW OF NICOL IN COMBINATION WITH KLING, DEHON, AND BHATIA UNDER BOTH PARTIES' CONSTRUCTIONS

266. In Section IX *supra*, I explained that Nicol in view of Kling and DeHon renders obvious Challenged Claims 1–6 under Petitioner's claim constructions. Specifically, regarding the claim 1 limitation requiring "the multiprocessor system setting a clock frequency . . . to a minimum . . ." ("to minimum" limitation), I explained that the limitation is met under Petitioner's construction of "setting a clock frequency ... to a minimum" to mean "setting the frequency to a level no more than is required for the preservation of memory contents or the like" even though Nicol, Kling and DeHon do not expressly disclose such a setting. *See* Section IX *supra*.

267. In my opinion, this is because setting a clock frequency “to a level no more than is required for the preservation of memory contents” corresponds to a power-saving mode that was not only well-known and prevalent in the computer industry, but it was also specified in the 1996 *Advanced Configuration and Power Interface* (ACPI) industry specification and described in detail in other patents and publications such as Bhatia. Ex. 1011, 1:6–17.
268. Underlying Section IX is my view that, even though the combination of Nicol in view of Kling and DeHon does not expressly disclose “setting the frequency to a level no more than is required for the preservation of memory contents or the like,” common knowledge in the art and the prevalent use of such a low power mode setting renders the “at minimum” limitation obvious.
269. However, in my opinion the Challenged Claims are also rendered obvious under the current Section X. The only difference between the discussion under Section IX, and the current Section X is that, under Section X, involves the *combination* of Nicol, Kling and Dehon *with Bhatia* for the limited reason that Bhatia describes the ACPI’s C3-state, a low power mode setting that satisfies the “at minimum” limitation under both Petitioner’s

construction, and thus satisfying Patent Owner's construction of plain and ordinary meaning.

270. For example, Bhatia teaches “setting the frequency to a level no more than is required for the preservation of memory contents or the like” (as required under Petitioner's construction of “at minimum) because, under the C3-state specified in ACPI and discussed in Bhatia, the processor clock is disabled and the processor does not perform any activity except maintenance of the cache memory contents. Further, in my opinion, the C3-state also satisfies Patent Owner's plain meaning construction of “to a minimum” because the clock frequency is zero (clock disabled).

271. Thus, in light of the analysis set forth under Sections VIII and IX, the Section X combination of Nicol, Kling, DeHon and Bhatia renders obvious all Challenged Claims under both parties' claim constructions.

272. Further, in my opinion, a POSITA would have been motivated to consult Bhatia because Nicol, Kling and Bhatia are all directed to interrelated thermal and power management aspects that are critical and ubiquitous in processor-based systems, and the techniques can all be used to control the operation of processors, such as DeHon's reconfigurable processors. Ex. 1011 (Bhatia), 1:6–7 (“Different types of power and thermal management techniques have been implemented in systems.”).

273. In addition, Bhatia describes in detail “low activity state[s] (e.g., deep sleep, stop grant, C1, C2, or C3 state)” a commonly used by power management systems to reduce power, but not described in much detail in Nicol or Kling. Ex. 1011, 12:58–64 (citing the *Advanced Configuration and Power Interface* (ACPI) industry standard specification).
274. Thus, in my opinion, a POSITA would have been motivated to consult Bhatia and implement those low activity states in Nicol’s system as additional means to minimize power consumption. In my opinion, a POSITA would have had a reasonable expectation of success in doing so because Bhatia explains that those low activity states were commonly used in the industry.

XI. SECONDARY CONSIDERATIONS OF NON-OBVIOUSNESS

275. It is my opinion that secondary considerations do not support the non-obviousness of the Challenged Claims. I have seen no evidence that supports any secondary considerations tending to show non-obviousness, including commercial success, long-felt need, failure of others, skepticism, praise, teaching away, recognition of a problem, or copying by competitors.
276. I am not aware of anything suggesting commercial success of any products can be attributed to the Challenged Claims.

277. I am also unaware of there being a long-felt need at the time of the alleged invention for an invention utilizing the elements recited in the Challenged Claims.
278. I am also not aware of any failure of others to design or implement an invention similar to the one recited in the Challenged Claims. In fact, I believe there are numerous similar systems, including but not limited to the ones described in above, which pre-date the Challenged Claims.
279. I have reviewed numerous prior art references from around the time of the alleged invention and am not aware of any skepticism, praise, or teaching away by others of the alleged invention recited in the Challenged Claims. In fact, the opposite is true: prior art references explain the ready combination of the cited prior art systems. I am likewise unaware of any recognition afforded to the alleged invention recited in the Challenged Claims.
280. I am not aware of any copying of the Challenged Claims for their designs or products.
281. Should the Patent Owner assert that secondary considerations support a finding of non-obviousness, I reserve the right to submit a Declaration addressing those new assertions.

XII. CONCLUSION

282. I hereby declare under penalty of perjury under the laws of the United States of America that the foregoing is true and correct, and that all statements made of my own knowledge are true and that all statements made on information and belief are believed to be true. I understand that willful false statements are punishable by fine or imprisonment or both. *See* 18 U.S.C. § 1001.

Date: February 7, 2020

Respectfully submitted,

A handwritten signature in blue ink, appearing to read "Carl Sechen", with a stylized, cursive script.

Dr. Carl Sechen