

Filed on behalf of OpenSky Industries, LLC

By: Andrew T. Oliver, Reg. No. 62,764
aoliver@ATWiplaw.com
Vinay V. Joshi, Reg. No. 51,034
vjoshi@atwiplaw.com
AMIN, TUROCY & WATSON LLP
160 W. Santa Clara Street
Suite 975
San Jose, CA 95113
Tel: (650) 618-6477

UNITED STATES PATENT AND TRADEMARK OFFICE
BEFORE THE PATENT TRIAL AND APPEAL BOARD

OpenSky Industries, LLC,
Petitioner,

v.

VLSI Technology LLC,
Patent Owner

Case IPR2021- 01064

**PETITION FOR *INTER PARTES* REVIEW OF
U.S. PATENT 7,725,759
CHALLENGING CLAIMS 1, 14, 17-18, 21-22, 24
UNDER 35 U.S.C. §312 AND 37 C.F.R. §42.104**

TABLE OF CONTENTS

I.	Introduction.....	5
II.	Mandatory Notices.....	5
A.	Real Party-In-Interest.....	5
B.	Related Matters	5
C.	Counsel	5
D.	Service Information	5
III.	Certification of Grounds for Standing	6
IV.	Overview of Challenge and Relief Requested	6
A.	Prior Art Patents and Printed Publications.....	6
B.	Grounds for Challenge.....	7
V.	Discretion Under 35 U.S.C. §314(a) and §325(d)	7
VI.	Brief Technology Description.....	10
A.	Clock Frequency and Power Consumption.....	10
B.	Controlling Components With a Clock Frequency Signal	11
C.	Adjusting Clock Speed to Balance Performance and Power	12
D.	Requests for Clock Speeds.....	12
E.	Arbiters	12
VII.	Overview of the ‘759 Patent	13
A.	The Alleged Problem	13
B.	The Alleged Invention	14
1.	Request.....	15

2.	Master Devices.....	15
3.	Predefined Change in Performance Due to Loading of the First Master Device.....	16
4.	Arbiter.....	16
C.	Summary of Prosecution History.....	16
VIII.	Claim Construction.....	17
IX.	Level of Ordinary Skill in the Art.....	17
X.	Prior Art Overview	17
A.	Chen (Ex. 1003).....	17
B.	Terrell (Ex. 1004).....	18
C.	Shaffer (Ex. 1005).....	19
D.	Lint (Ex. 1006).....	20
E.	Kiriake (Ex. 1028)	21
XI.	Specific Grounds.....	22
A.	Ground I: Claims 1, 14, 17 are Obvious Over Shaffer in View of Lint.....	22
1.	Claim 1.....	22
2.	Claim 14.....	31
3.	Claim 17.....	33
B.	Ground II: Claims 18, 21-22, 24 are Obvious Over Shaffer in View of Lint and Kiriake	34
1.	Claim 18.....	34
2.	Claim 21	38
3.	Claim 22.....	38
4.	Claim 24.....	39
C.	Ground III: Claims 1, 14, 17 are Obvious Over Chen in View of Terrell.....	40

1.	Claim 1	40
2.	Claim 14.....	49
3.	Claim 17	52
D.	Ground IV: Claims 18, 21-22, 24 are Obvious Over Chen in View of Terrell and Kiriake.....	54
1.	Claim 18.....	54
2.	Claim 21	59
3.	Claim 22.....	60
4.	Claim 24.....	60
XII.	Conclusion	61

I. Introduction

OpenSky Industries, LLC (“OpenSky” or “Petitioner”) respectfully requests *inter partes* review of claims 1, 14, 17-18, 21-22, 24 of U.S. Patent 7,725,759 (the “‘759 patent”) (Ex. 1001) pursuant to 35 U.S.C. §§ 311-319 and 37 C.F.R. § 42.1 *et seq.*

II. Mandatory Notices

A. Real Party-In-Interest

OpenSky Industries, LLC is the real party-in-interest.

B. Related Matters

Upon information and belief, VLSI Technology LLC (“VLSI” or “Patent Owner”) asserted the ‘759 patent against Intel Corporation in *VLSI Technology LLC v. Intel Corporation*, Case No. 6:19-cv-00254-ADA (consolidated as 19-cv-00977) (W.D. Tex.). A jury verdict was rendered in that case. Ex. 1027.

C. Counsel

Lead Counsel: Andrew T. Oliver (Reg. No. 62,764)

Backup Counsel: Vinay V. Joshi (Reg. No. 51,034)

D. Service Information

Email: aoliver@atwiplaw.com

vjoshi@atwiplaw.com

Mail and hand delivery:

Amin, Turocy & Watson LLP

160 W. Santa Clara St.

Suite 975

San Jose, CA 95113

Telephone: 650-618-6477

Fax: N/A

Petitioner consents to email delivery on lead and backup counsel.

III. Certification of Grounds for Standing

Petitioner certifies pursuant to Rule 42.104(a) that the patent for which review is sought is available for *inter partes* review and that Petitioner is not barred or estopped from requesting an *inter partes* review challenging the patent claims on the grounds identified in this Petition.

IV. Overview of Challenge and Relief Requested

Pursuant to Rules 42.22(a)(1) and 42.104(b)(1)-(2), Petitioner challenges claims 1, 14, 17-18, 21-22, 24 of the '759 Patent.

A. Prior Art Patents and Printed Publications

The application for the '759 patent was filed before the America Invents Act ("AIA").

The following references are pertinent to the grounds of unpatentability:

1. U.S. Patent 5,838,995 ("Chen") (Ex. 1003), issued November 17, 1998, is prior art under at least 35 U.S.C. §102(b).
2. U.S. Patent Application Publication No. 2004/0098631 ("Terrell") (Ex. 1004), published May 20, 2004, is prior art under at least 35 U.S.C. §102(b).
3. U.S. Patent Application Publication No. 2003/0159080 ("Kiriake") (Ex. 1028), published August 21, 2003, is prior art under at least 35 U.S.C. §102(b).
4. U.S. Patent 6,298,448 ("Shaffer") (Ex. 1005), issued October 2, 2001, is prior art under at least 35 U.S.C. §102(b).
5. U.S. Patent 7,360,103 ("Lint") (Ex. 1006), filed on May 21, 2004, and issued on April 15, 2008, is prior art under at least 35 U.S.C. §102(e).

B. Grounds for Challenge

Petitioner requests cancellation of claims 1, 14, 17-18, 21-22, 24 of the '759 patent as unpatentable under 35 U.S.C. §103. This Petition, supported by the declarations of Dr. Bruce Jacob (Exs. 1002, 1046), demonstrates a reasonable likelihood that Petitioner will prevail with respect to cancellation of at least one challenged claim and that each challenged claim is not patentable. See 35 U.S.C. §314(a).

The grounds for challenge are:

	References for §103 Grounds	Challenged Claims
1.	Shaffer, Lint	1, 14, 17
2.	Shaffer, Lint, Kiriake	18, 21-22, 24
3.	Chen, Terrell	1, 14, 17
4.	Chen, Terrell, Kiriake	18, 21-22, 24

V. Discretion Under 35 U.S.C. §314(a) and §325(d)

OpenSky is not Intel. OpenSky has never been accused of infringing the challenged patent. There is no basis for denying institution of an otherwise meritorious petition based on infringement litigation between the patent owner and a third party. Moreover, it appears the jury was not presented with any invalidity grounds. Cases such as this, “do not warrant denial.” *Wells Fargo Bank, N.A. v. USAA*, IPR2019-01082, Paper 9, pp. 13-14 (PTAB Dec. 13, 2019). The patent owner may argue for a discretionary denial, and if the Board were to consider the state of the litigation between the patent owner and third-party Intel relevant (though it is not), the *Fintiv* factors show that OpenSky’s meritorious petition should move forward.

The first *Fintiv* factor¹ – court granted stay – is inapplicable, as no lawsuit is pending regarding OpenSky. And the VLSI v. Intel trial already occurred.

The second *Fintiv* factor – proximity of trial date – is inapplicable as to OpenSky and might nominally weigh against institution with respect to VLSI v. Intel, except that the grounds presented in this petition were seemingly never presented to the district court. That irrelevant trial concluded before this petition was even filed. As discussed further below, it does not appear that the district court ever spent any resources considering the invalidity of this patent.

The third *Fintiv* factor – investment in the parallel proceeding – weighs in favor of institution, as no lawsuit has been filed against OpenSky. With respect to the irrelevant VLSI v. Intel proceeding, OpenSky is not a party to the litigation for this patent and cannot fully know what investment has been made by either party regarding the invalidity grounds presented in this petition. But OpenSky reviewed the extensive PACER dockets for the VLSI v. Intel cases where the ‘759 patent was asserted and did not see any summary judgment motions or Daubert motions regarding invalidity among the many filed motions. If OpenSky overlooked such a motion, OpenSky assumes that patent owner will correct the record in its response. In addition, it appears that the court did not construe any terms from the challenged claims. *See* Ex. 1045 p. 2. And, more importantly, the jury was not asked to consider the invalidity of this patent. Ex. 1027. Factor three weighs in favor of institution.

The fourth *Fintiv* factor – overlap between issues in petition and court – weighs in favor of institution. As noted, it does not appear that invalidity was ever presented to judge or jury in the VLSI v. Intel lawsuit. Moreover, this petition addresses more claims than the district court trial. The trial related to claims 14, 17-18, and 24 of the ‘759 patent. *See* Ex. 1027 p. 3. This petition

¹ *Apple Inc. v. Fintiv, Inc.*, IPR2020-00019, Paper 11 (PTAB Mar. 20, 2020)

further addresses claims 1, 21, and 22. Where the petition addresses more claims than were at issue during trial, this factor weighs in favor of institution. *See, Apple, Inc. v. Maxell Ltd.*, IPR2020-00204, Paper 11, p. 17 (PTAB June 19, 2020).

The fifth *Fintiv* factor – whether petitioner and defendant in parallel proceeding are the same party – weighs in favor of institution. OpenSky is not Intel. OpenSky has no relationship to Intel. OpenSky is not a party, privy, or RPI to any party alleged to infringe the ‘759 patent in any district court proceeding. “If a petitioner is unrelated to a defendant in an earlier court proceeding, the Board has weighed this fact against exercising discretion.” *Id.* (quoting *Fintiv*).

The sixth *Fintiv* factor – other circumstances, including the merits – weighs in favor of institution. At the time of making previous decisions whether to institute review of the ‘759 patent, the Board had the unenviable task of peering a year into the future to determine whether a patent trial in a busy district court would go forward on time or be delayed. The current task is different. The district court trial is finished and, based on the verdict form, the Board knows exactly what the judge and jury were asked to decide. *See* Ex. 1027. Invalidity was not even considered. Thus, the Board knows that its decision will not create any inefficiency or risk an inconsistent decisions.

Here, the Board needs to institute review to maintain the integrity of the patent system, because a jury found that this patent is worth at least \$675 million (\$675,000,000), yet no judge or jury (or PTAB proceeding) has ever double-checked the validity of the ‘759 patent. The *Fintiv* analysis is designed to determine whether the integrity of the system would be furthered by instituting review. *Apple v. Fintiv*, IPR2020-00019, Paper 11, p. 6 (“the Board takes a holistic view of whether efficiency and integrity of the system are best served by denying or instituting review.”). The integrity of the entire patent system is threatened whenever a patent owner constructs a set of proceedings in which no one ever checks the validity of a patent found to be worth over six hundred million dollars. The denial of invalidity review cannot be proper; OpenSky urges the Board to find

that this factor weighs strongly in favor of institution.

Thus, nearly every *Fintiv* factor favors institution. This petition should not be denied based upon prior litigation between VLSI and Intel. This petition is brought by a different and unrelated petitioner; more claims are at issue in this petition than in litigation; the invalidity grounds presented here have never been considered by a district court judge or jury.

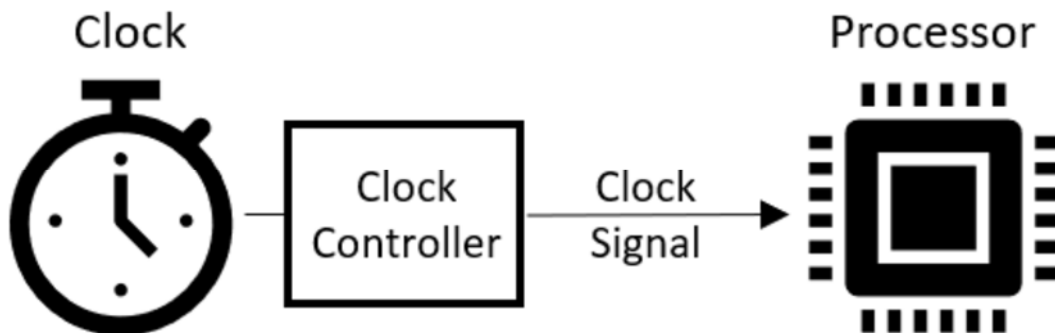
OpenSky urges the board to grant review according to the substantive merit of this petition.

VI. Brief Technology Description

The '759 patent relates to managing clock frequency in an electronic device.

A. Clock Frequency and Power Consumption

Computing devices typically include processors executing instructions at a certain frequency, or speed. A clock in a computing device typically generates a fixed sinusoidal signal corresponding to the oscillation frequency of a crystal. As illustrated below, a clock controller may vary a clock's speed and send the resulting clock signal to control the frequency at which a component (a processor) operates. Ex. 1046 ¶¶42-43; Ex. 1002 ¶¶42-43.



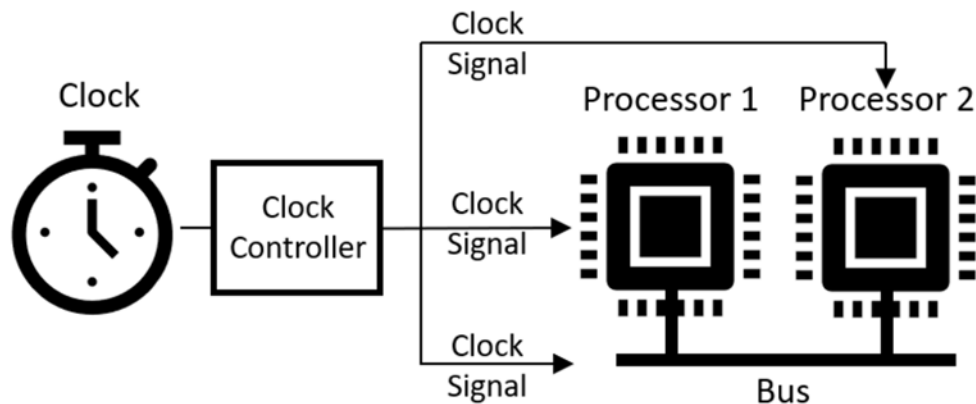
Decades before the June 29, 2005 filing date of the '759 patent ("Filing Date"), clock controllers providing variable frequencies were well-known. For example, Sheets, filed 1984, discloses a voltage-controlled oscillator that "adjusts the frequency of the clock signal transmitted to microprocessor...." Ex. 1008, 2:65-67; *see* Exs. 1007, 1029. Ex. 1046, ¶¶27, 51; Ex. 1002 ¶¶27, 51.

Increasing clock frequency increases the speed of components (processors) running on that clock. Increased performance increases power consumption. Generally, decreasing clock frequency decreases both performance and power consumption, whereas increasing the clock frequency increases both. This has long been known. The '759 patent admits: "as the clock frequency increases to deliver more performance, the power consumption ... also increases." Ex. 1001, 1:19-21; Ex. 1002, ¶¶60-65, 72; Ex. 1046, ¶¶65-70, 77.

B. Controlling Components With a Clock Frequency Signal

Before the Filing Date, it was known that systems could include many components, or "devices," and that some devices could be "master" devices. The '759 patent states that a "master device" is one that "may initiate communication." Ex. 1001, 3:13-15. A master device may be "a processor, an input/output bus controller, a direct memory access (DMA) controller, an error correction code module or an external memory interface." *Id.*, 3:23-25. Before the Filing Date, "master devices" were well-known. Ex. 1003, 4:42-43 ("If any of the devices are either the controlling master...."). Ex. 1002, ¶¶29-35, 53-55, 71; Ex. 1046, ¶¶29-35, 53-55, 76.

It was also well-known that devices communicated over a "bus." Ex. 1039, 346 ("using a standard bus structure"). A simple bus is one or more conductors (wires) over which device-to-device communications pass. Ex. 1002 ¶¶53-59; Ex. 1046, ¶¶53-59.



As shown above, a clock controller may provide a clock signal not only to master devices, but also to a bus. This was known long before the Filing Date. Ex. 1004, ¶¶0005-0008 (“An on-chip bus that hosts two or more bus masters, all of which share a common bus clock.”); Ex. 1005, 1:46-51 (“[O]ften the entire system bus operates off of a single clock...”). Ex. 1002, ¶¶45-46, 92, 111; Ex. 1046, ¶¶45-46, 114, 142.

C. Adjusting Clock Speed to Balance Performance and Power

It was known long before the Filing Date that power consumption is directly proportional to frequency (speed). It was known, therefore, that performance should be balanced with power conservation, and that always running at full speed was not desirable. Ex. 1008, 1:31-38; see Exs. 1035-1038²; Ex. 1002, ¶¶27, 65-70; Ex. 1046, ¶¶27, 65-70.

One prior-art technique for balancing performance with power consumption was to vary clock speed based on device workload. For example, “[u]pon determining the amount of processing required,” a microprocessor could “compute[] an operating frequency that is sufficient to meet the offered processing load” and “gradually adjust[] the frequency of the clock signal ... to the computed frequency....” Ex. 1008, 2:58-67. Ex. 1002, ¶¶27, 65-70, 110; Ex. 1046, ¶¶27, 65-70, 93.

D. Requests for Clock Speeds

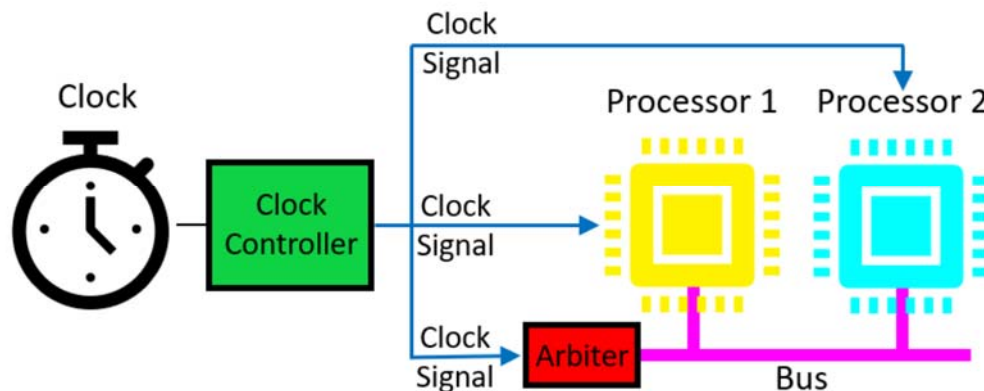
It was known before the Filing Date that devices could “request” changes in clock speed. Ex. 1003, 3:3-17 (“Each of devices 34 and 36 have speed requesting circuits.... The speed request circuits ... each output a control signal to clock gate logic circuit 24 ... [to] optimize the frequency of data communications between devices 36 and 34....”). Ex. 1002, ¶¶77-78; Ex. 1046, ¶¶77-78.

E. Arbiters

It was known before the Filing Date that an arbiter controls the flow of data on a bus.

² These exhibits and Ex. 1009 are publicly available prior art. See Ex. 1040.

When multiple master devices connect to a bus, an arbiter has traditionally been used to arbitrate the multiple master devices initiating transactions on the bus simultaneously, to prevent system lockups, instability, or data loss. In many devices, an arbiter is integrated into a Peripheral Component Interconnect (“PCI”) chipset such as a PCI host bridge, or peripheral bus controller. As shown below, an arbiter coupled to the bus and processors can control the direction and/or speed of data flow on the bus, to and from each processor. Centrally arbitrating bus transactions with an arbiter was known since the 1980s. Ex. 1041, Fig. 1; Ex. 1042, Fig. 3; Exs. 1043-1044 at Fig. 2; Ex. 1046, ¶¶53-64.



VII. Overview of the '759 Patent

A. The Alleged Problem

The '759 patent acknowledges a well-known problem: devices must perform tasks “quickly and efficiently,” and one way to “increase the performance” was to “increase the clock frequency of the clock used in the device.” Ex. 1001, 1:10-24. However, “as the clock frequency increases to deliver more performance, the power consumption ... also increases” because “power consumption is proportional to the number of transitions on the logic.” *Id.*, 1:10-24, 3:46-48. Ex. 1002, ¶72; Ex. 1046, ¶77.

B. The Alleged Invention

To “selectively deliver faster clock speeds,” the ’759 patent claims three purportedly novel concepts: (1) a “first master device” providing a “request ... to change a clock frequency of a high-speed clock”; (2) the request being provided “in response to a predefined change in performance of the first master device ... due to loading of the first master device as measured within a predefined time interval”; and (3) responsive to the request, providing the clock frequency of the high-speed clock to both “a second master device” and a “bus.” Ex. 1001, Claims 1, 14, 18. All were well-known before the Filing Date. Ex. 1002, ¶¶27-72, 75-120; Ex. 1046, ¶¶27-127.

Annotated Figure 1 of the ’759 patent, below, illustrates elements from claims 1, 14, and 18. It shows first master device (120) and second master device (122) coupled to bus (102) and arbiter (110). Ex. 1001, 2:67-3:3. It also shows first master device (120) sending a request to change a clock frequency, and clock controller (150) outputting a high-speed clock (152). Id., 2:58-3:38. Ex. 1002 ¶¶69-70; Ex. 1046, ¶¶74-77.

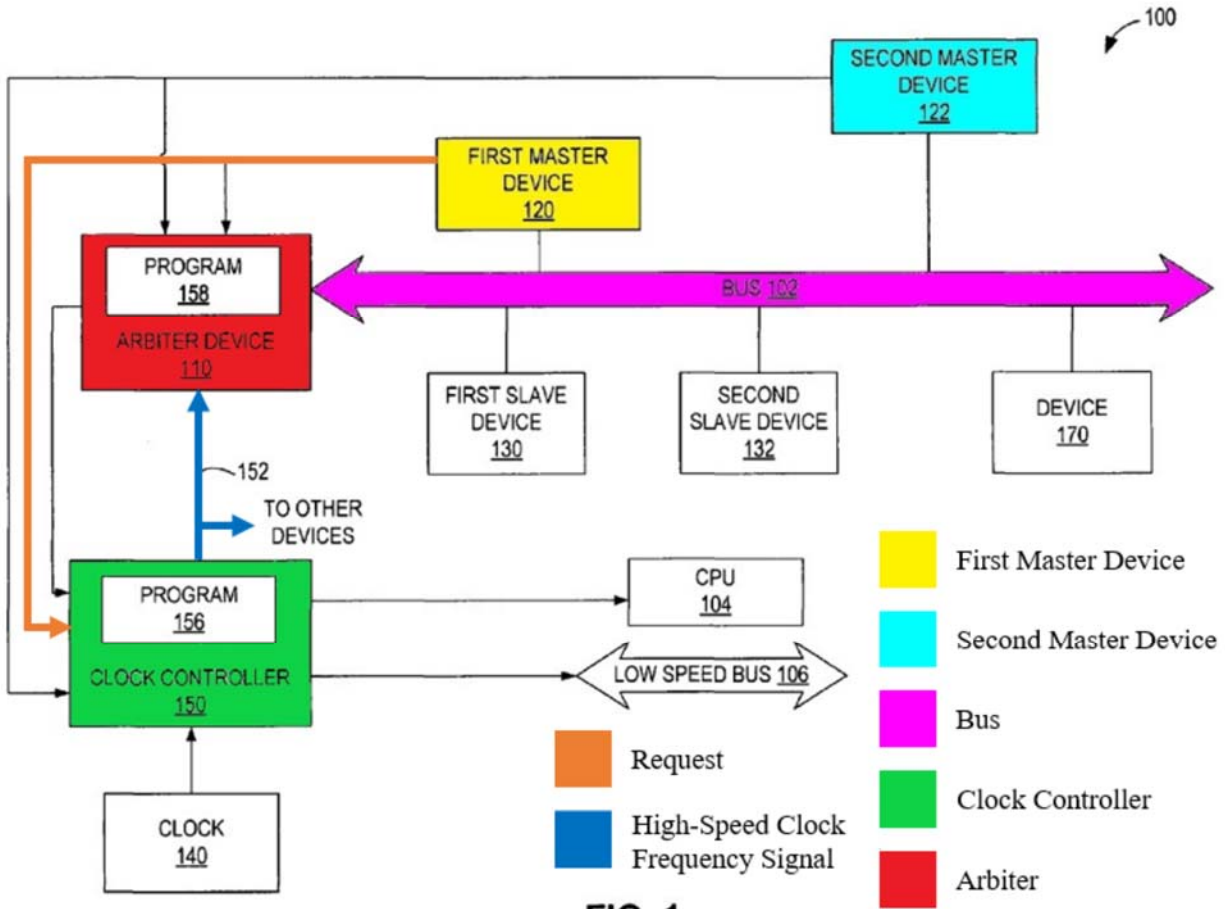


FIG. 1

1. Request

The claims require that a “first master device” send a “request” “to change a clock frequency of a high-speed clock.” Ex. 1001, Claims 1, 14, 18. The ’759 patent refers to “requests” as “trigger signals,” and explains that “clock controller 150 controls and/or adjusts the high speed clock 152 by changing the clock frequency in response to the plurality of trigger signal inputs.” *Id.*, 4:6-29; Ex. 1002 ¶¶69-70; Ex. 1046, ¶¶74-77.

2. Master Devices

The claims require a “first master device” and “second master device.” Ex. 1001, Claims 1, 14, 18. A master device “may initiate communication ... by requesting an access token from the arbiter 110 to communicate over the bus 102.” *Id.*, 3:13-15. Examples include: “a processor, an

input/output bus controller, a direct memory access (DMA) controller, an error correction code module or an external memory interface.” *Id.*, 3:23-25; Ex. 1002 ¶¶69, 71; Ex. 1046, ¶¶74-77.

3. **Predefined Change in Performance Due to Loading of the First Master Device**

The claims require that the request to change a high-speed clock frequency signal be “in response to a predefined change in performance of the first master device, wherein the predefined change in performance is due to loading of the first master device as measured within a predefined time interval.” Ex. 1001, Claims 1, 14, 18. This loading “may include a level of audio processing or signal output of an MP3 player.” *Id.*, 3:64-4:6; Ex. 1002 ¶¶69-70; Ex. 1046, ¶¶74-77.

4. **Arbiter**

Claims 18, 21-22, and 24 require “an arbiter coupled to the bus and coupled to the first master device” and coupled to the clock controller, “the arbiter configured to control flow of data on the bus.” Ex. 1001, Claim 18. The arbiter controls data flow on the bus by granting access tokens to master devices to permit bus communication. *Id.*, 3:11-15; Ex. 1046, ¶¶74-79.

C. **Summary of Prosecution History**

The ’759 patent issued from Application No. 11/170,475 (Ex. 1010). The original claims claimed monitoring a plurality of master devices coupled to a bus and receiving a request to increase the clock frequency of the bus. Ex. 1010, Specification, 14-18.

Before any office action, the applicant amended several claims. See Ex. 1011, 2-6. The Office rejected the claims seven times, leading to seven amendments, including a replacement claim set with an RCE dated September 10, 2008. Exs. 1012-1025.

In addition, the applicant argued distinctions over several prior-art references. Exs. 1030-1034. For example, the applicant argued that U.S. Patent 7,007,121 (Ex. 1033) failed to disclose the limitation “input ... to request an increase to the clock frequency of the bus” because it purportedly “does not specify that the master device identifies an appropriate frequency.” Ex. 1013, 9. The

applicant argued that a request from a device that “happens to” increase the bus speed is not a “request to change the bus speed.” See Ex. 1019, 10.

After several amendments, the claims were allowed. Ex. 1026.

VIII. Claim Construction

In an *inter partes* review, a patent claim “shall be construed using the same claim construction standard that would be used to construe the claim in a civil action ..., including ... the ordinary and customary meaning ...” understood by a person of ordinary skill in the art (“POSA”). 37 C.F.R. § 42.100(b). Each claim term should be construed in accordance with its ordinary and customary meaning, and no terms otherwise need construction.

IX. Level of Ordinary Skill in the Art

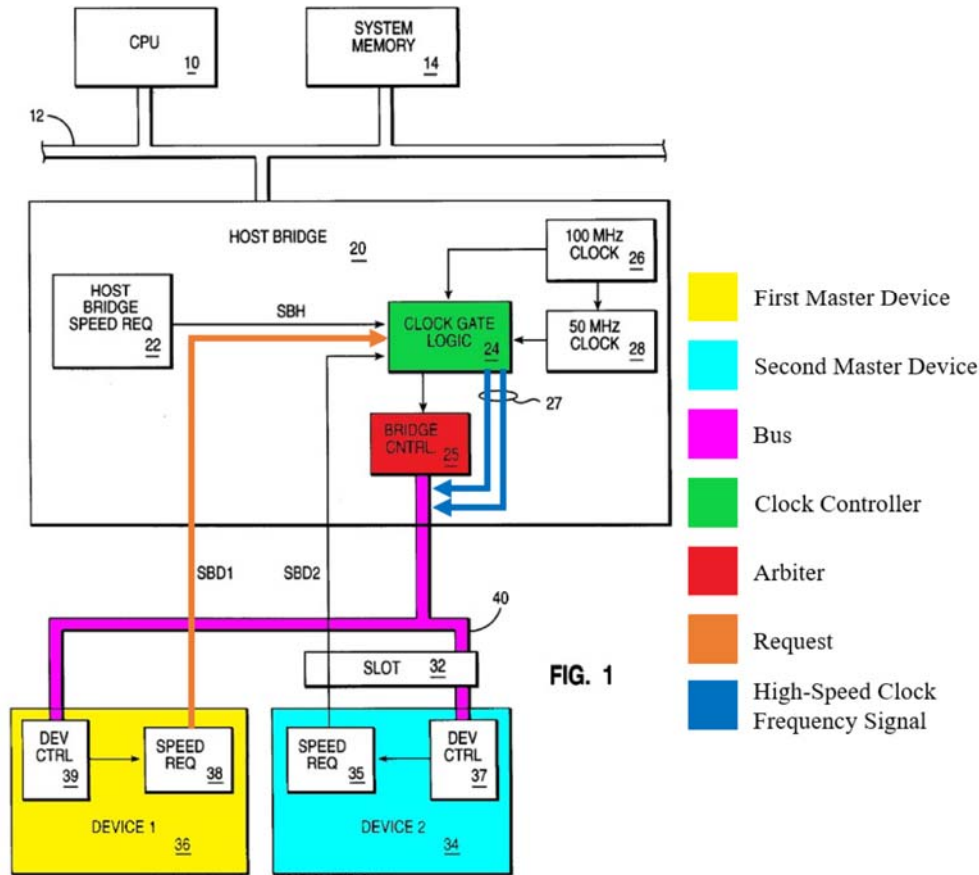
A POSA at the time of the alleged invention would have had at least the following (or equivalents): a B.S. degree in computer science or electrical engineering with at least five years of experience in computer architecture design; or an M.S. degree in computer science or electrical engineering with at least two years of experience in computer architecture design. Ex. 1002, ¶26. Ex. 1046, ¶26

X. Prior Art Overview

Prior art disclosed all elements of the challenged claims—including the purportedly novel features—before the Filing Date. Ex. 1046, ¶¶80-127 Ex. 1002, ¶¶75-120.

A. Chen (Ex. 1003)

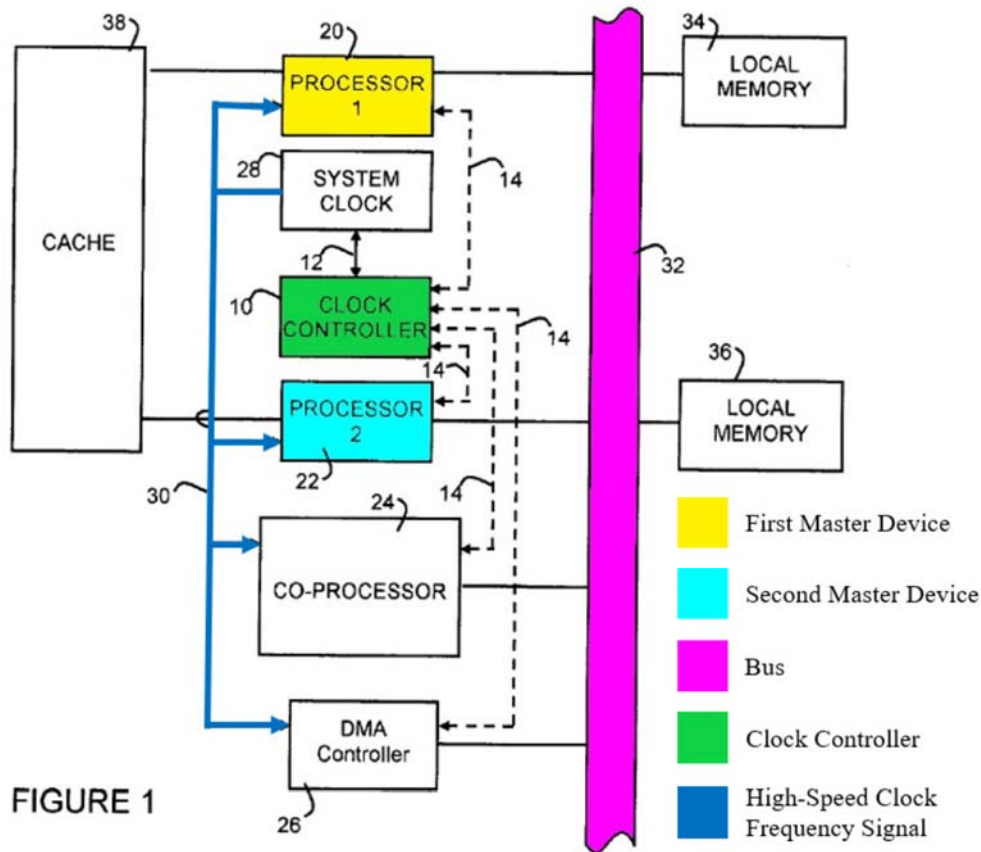
Chen discloses a “data processing system” with two “devices” 34, 36 coupled to “bus” 40, clock gate logic circuit 24, and bridge control logic 25. Ex. 1003, Fig. 1; Ex. 1002 ¶¶75-79; Ex. 1046, ¶¶80-87.



Chen explains that certain devices might be capable of operating at faster speeds than others. Ex. 1003, 1:13-32. A CPU might be capable of operating at a higher frequency than Input/Output (I/O) devices. *Id.* Chen proposes “control logic ... which allows the bridge to perform bus transactions at both a high frequency and a lower frequency.” *Id.*, 2:1-8. If a device “is capable of operating at a higher frequency, then a control signal is transmitted” from the device, resulting in the “host bridge, bus and I/O device ... all then operating at the higher frequency.” *Id.*, 2:6-14; Ex. 1002 ¶¶80-82, 123; Ex. 1046, ¶¶80-87.

B. Terrell (Ex. 1004)

Terrell discloses at least two processors 20, 22 coupled to bus 32 and clock controller 10. Ex. 1004, Fig. 1 (annotated below). Ex. 1002 ¶¶105-108; Ex. 1046, ¶¶88-94.



Terrell explains, “Power management of chips ... by adjusting the system clock frequency is [] well-known.... The system clock can be adjusted downward or upward to match the processing requirements.” Ex. 1004, ¶¶0002. Terrell thus teaches “measuring processor clock usage amongst multiple processing elements that share a common clock,” and determining a frequency for the common clock. *Id.*, ¶¶0009. Systems “where such a control feature would be desirable include: ... An on-chip bus that hosts two or more bus masters, all of which share a common bus clock.” *Id.*, ¶¶0006-0008; Ex. 1002 ¶¶106-111; Ex. 1046, ¶¶88-94.

C. Shaffer (Ex. 1005)

Shaffer discloses one or more master devices (CPU 20, controllers 22, 24), bus 21, and programmable clock module 50 that provides a clock frequency signal (“a programmable variable clock frequency”) to CPU 20 and “to the other controllers and buses ...” Ex. 1005, 4:15-21, Fig. 1

(annotated below). Shaffer discloses other master devices, i.e., multiple processors. *Id.*, 6:2-5. Ex.

1002 ¶¶89-91; Ex. 1046, ¶¶111-118.

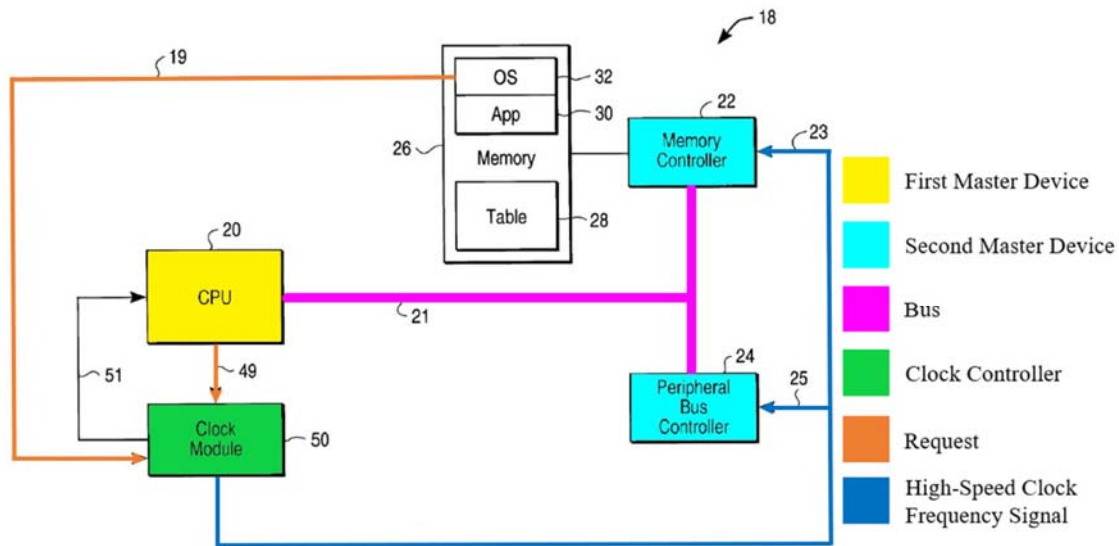


FIG. 1

Shaffer proposes a “CPU speed control system” “for varying the CPU clock rate to correspond with the processing requirements of the device.” Ex. 1005, 2:13-20, 2:46-49, 2:63-67. “CPU 20 ... can instruct through line 49 the clock module 50 to increase or decrease the output frequency ... enabling the CPU 20 to regulate its own operating clock speed.” *Id.*, 3:14-22. Clock frequency may be changed “in response to a CPU utilization application that dynamically monitors the level of CPU usage.” *Id.*, 4:50-54. The resulting clock speed “[t]ypically ... drives the entire system bus ... and thereby reduces power requirements for the processor, related chipsets, memory, controllers ...” *Id.*, 5:65-6:2. Ex. 1002 ¶¶93-95; Ex. 1046, ¶¶111-118.

D. Lint (Ex. 1006)

Lint discloses processors (670, 680) and “bus 616 (a PCI bus).” Ex. 1006, 9:30-32, 9:53-54, Fig. 6 (annotated below). Ex. 1002 ¶83; Ex. 1046, ¶¶105-110.

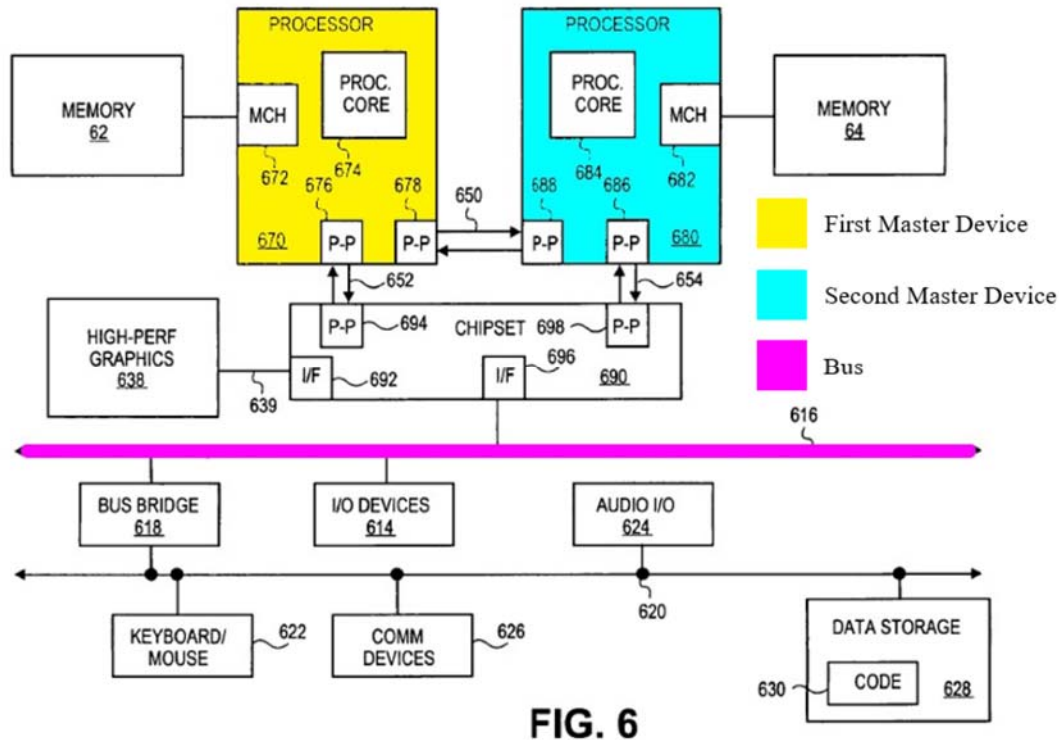


FIG. 6

Lint describes “changing both the operating voltage and frequency of the processor such that transition to a lower frequency (performance) point leads to a cubic reduction in power consumption.” Ex. 1006, 1:21-25. An operating system (“OS”) controls “so that the processor is placed at an operating frequency that matches the processor utilization.” *Id.*, 1:26-34. Ex. 1002 ¶¶83-85; Ex. 1046, ¶¶105-110.

E. Kiriake (Ex. 1028)

Kiriake discloses two Peripheral Processors (10, 11—master devices described in Section VII.B.2), clock controller 3, system bus 12, and arbiter 2 “that performs arbitration relating to use of this system bus.” Ex. 1028, ¶0011, Fig. 1 (annotated below). “The arbiter 2 arbitrates the bus request made through the bus request signal (BUSREQ). The arbiter 2 then asserts the bus grant signal(GRANT) to the CPU 1 or peripheral processor 10 or 11 that is recognized as the bus master as a result of arbitration.” *Id.*, ¶0027. Ex. 1046, ¶¶95-99.

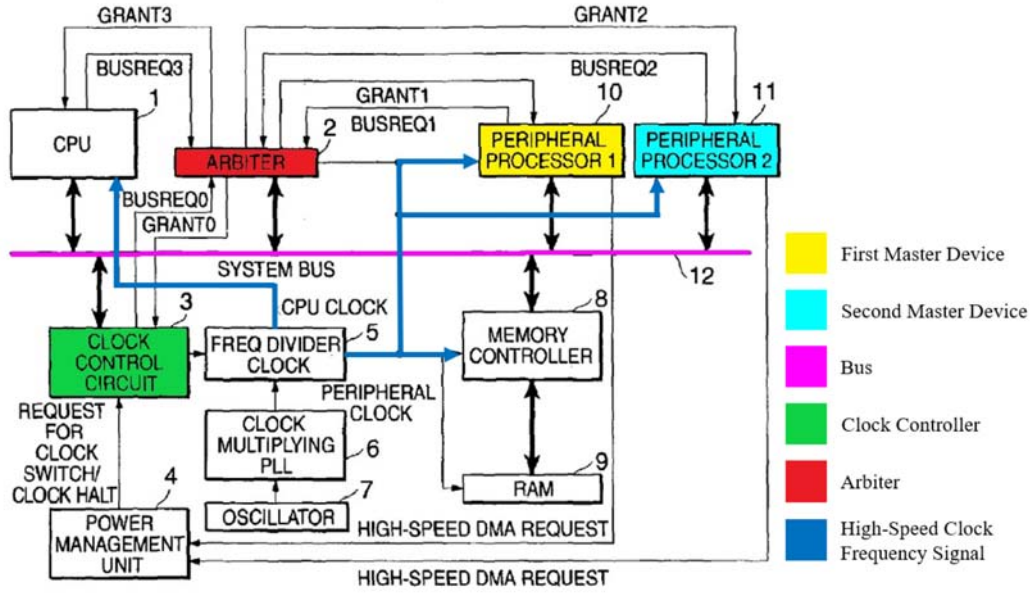


FIG. 1

XI. Specific Grounds

A. Ground I: Claims 1, 14, 17 are Obvious Over Shaffer in View of Lint

1. Claim 1

- (a) “A method comprising: monitoring a plurality of master devices coupled to a bus”

Shaffer teaches a plurality of master devices (CPU 20, a memory controller 22, a bus controller 24, and another CPU) coupled to a bus (data/command bus 21).

See Ex. 1005, Fig. 1

(annotated), 6:2-5. Ex.

1002, ¶¶228-229.

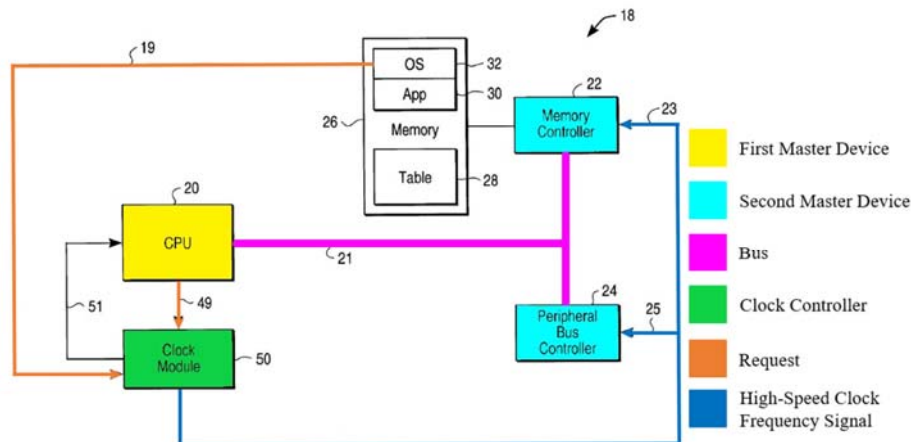


FIG. 1

A “master device” may be “a processor, an input/output bus controller, a direct memory access (DMA) controller, an error correction code module or an external memory interface.” Ex.

1001, 3:23-28. Shaffer's CPUs, "memory controller," and "peripheral bus controller" are such master devices. Ex. 1005, 4:18-19. A POSA would have understood that Shaffer's "controllers" could initiate communications, like those of the '759 patent. *See* Ex. 1001, 3:12-15. Shaffer discloses multiple CPUs. Ex. 1005, 6:2-5 (multiple processors), Claims 1, 3-6, 8, 16, 18, and 19 ("one or more CPUs" and buses). These CPUs are master devices, per the '759 patent. Ex. 1002, ¶¶229-233.

Shaffer illustrates that at least one CPU, as well as the memory controller and bus controller—the master devices—are coupled to a bus 21. Ex. 1005, Fig. 1, 5:66-6:2, 4:23-25. A POSA would have found it obvious that the other CPUs disclosed by Shaffer would have been coupled to the bus. Ex. 1002, ¶¶228-233.

Shaffer teaches a plurality of master devices coupled to a bus. Shaffer teaches monitoring the master devices with "a CPU utilization application that dynamically monitors the level of CPU usage." Ex. 1005, 4:53-54, 4:50-5:20. Shaffer teaches clock module 50 monitoring instructions received, e.g., via lines 49 and/or 19. *Id.*, 3:16-18, 3:53-56, Figure 1.

Shaffer teaches this limitation. Ex. 1002, ¶234.

(b) "receiving a request, from a first master device of the plurality of master devices, to change a clock frequency of a high-speed clock"

Shaffer discloses receiving a request (instructions via lines 19 and 49) from a first master device (CPU 20) to change a clock frequency of a highspeed clock ("the output frequency" signal of clock module 50). A "CPU control system 18" includes a "CPU 20 coupled to an intelligent programmable clock module 50." Ex. 1005, 3:8-22. The clock module 50 has a "frequency synthesizer for providing the CPU 20 with a clocking signal." *Id.* "[T]o ensure the proper operating frequency of the CPU 20," Shaffer describes "a feedback circuit," in which "the clock module 50 ... inform[s] the CPU 20 of its output frequency through line 51, and the CPU 20 in turn can instruct through line 49 the clock module 50 to increase or decrease the output frequency as needed." *Id.*

This “enabl[es] the CPU 20 to regulate its own operating clock speed.” *Id.* This “instruct[ion] ... to increase or decrease the output frequency” from Shaffer’s CPU is a “request” from a first master device to change a clock frequency of a high-speed clock. Ex. 1002, ¶¶235, 241-242.

Additionally, “the OS 32 is used to control the frequency of the clock module 50 in response to a CPU utilization application that dynamically monitors the level of CPU usage.” Ex. 1005, 4:50-54. Shaffer explains, “Depending upon the CPU utilization values provided by the CPU monitor, the OS 32 generates an interrupt to the clock module 50 instructing it to raise or lower the clocking frequency provided to the CPU 20.” *Id.*, 5:5-20. Figure 1 shows this communication over line 19. *Id.*, 3:53-56. OS control is consistent with the ’759 patent’s explanation that functionality may be “implemented as hardware ... firmware ... and/or software or a combination thereof.” Ex. 1001, 2:51-57. Ex. 1002, ¶¶235, 239, 241-242.

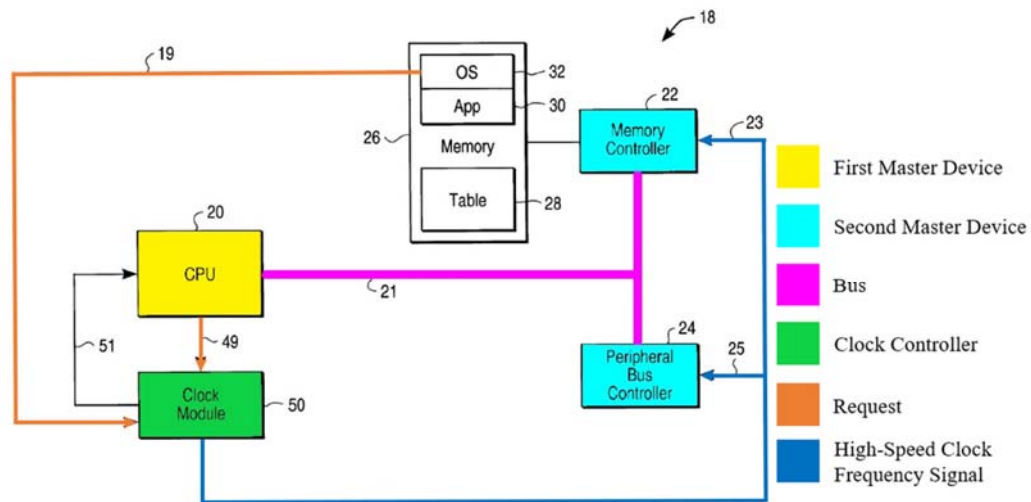


FIG. 1

Both Shaffer’s CPU 20 and the OS 32 may request a clock frequency: “[t]he operating system of the device and the CPU are both capable of controlling the frequency synthesizer output...” Ex. 1005, Abstract, 4:51-52. Ex. 1002, ¶¶235-237, 241-242.

Shaffer depicts “operating system (OS) 32, stored in a memory 26;” however, Shaffer does not expressly state which hardware executes the OS. Ex. 1005, 3:33-36. A POSA would have found

it obvious that a CPU executed the OS. A POSA would have understood that the requests from both the OS and the CPU in Shaffer came from CPU 20 (a master device). Ex. 1002, ¶¶235-237, 241-242.

Lint describes applications and an OS that execute on a processor. It was well-known that a system may include “an operating system (OS) 403 ..., firmware 404, and a processor 405.” Ex. 1006, 5:57-63. This well-known configuration is depicted below. *Id.*, Fig. 4 (annotated below); Ex. 1002, ¶¶217-218.

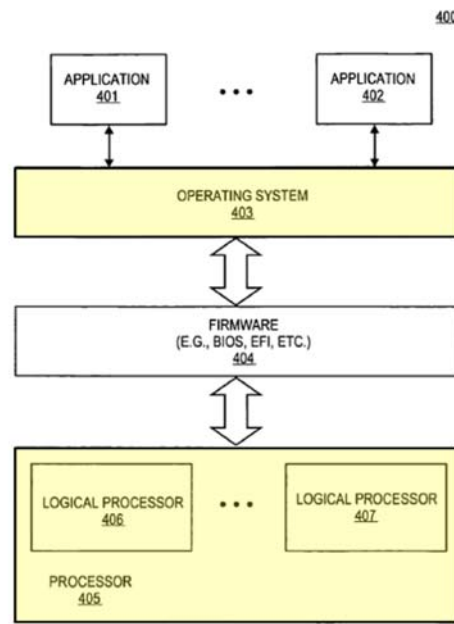


FIG. 4

This is consistent with Shaffer explaining that applications execute on CPUs. *See* Ex. 1005, Fig. 1, 4:51-5:20 (explaining that CPU monitoring programs measure CPU utilization levels that vary “depending on the particular application and the number of applications being executed at any given time” and describing “CPU executing the word processing program”). It was well-known and obvious that both Shaffer’s applications and OS would have executed on CPUs. Ex. 1002, ¶¶234-238.

In sum, Shaffer teaches receiving requests (instructions via lines 19 and 49 to clock module 50) from a first master device (CPU 20, executing OS 32 and applications 30) to change a clock frequency of a high-speed clock (the output signal of clock module 50). Ex. 1002, ¶¶234-238, 241-242.

(c) “the request sent from the first master device in response to a predefined change in performance of the first master device”

Shaffer discloses or renders obvious this limitation. As explained, §XI.A.1(b), a CPU can send an instruction (request) “through line 49” to “the clock module 50” to increase or decrease the output frequency as needed....” Ex. 1005, 3:15-18. The OS (32) “generates an interrupt to the clock module 50” instructing it to raise or lower the clocking frequency provided to the CPU 20.” Ex. 1005, 5:5-20. Ex. 1002, ¶¶235, 239-242.

Shaffer sends these instructions (requests) in response to a predefined change in performance of the first master device. The “frequency synthesizer output may be dynamically varied ... based on total CPU usage.” Ex. 1005, Abstract; 1:59-2:20; 2:45-52; 2:62-67. A Shaffer embodiment adjusts the clock speed based on the CPU utilization percentage. *Id.*, 4:50-5:30. There, “the OS 32 is used to control the frequency of the clock module 50 in response to a CPU utilization application that dynamically monitors the level of CPU usage.” *Id.* “CPU utilization levels vary depending on the particular application and the number of applications being executed at any given time.” *Id.* This may use 5-90% of the processing. *Id.* “Depending upon the CPU utilization values provided by the CPU monitor, the OS 32 generates an interrupt to the clock module 50 instructing it to raise or lower the clocking frequency provided to the CPU 20.” *Id.* And “[o]ptionally, the OS 32 may be programmed such that an interrupt is generated only when a predetermined threshold is exceeded.” *Id.* “Similarly, as CPU utilization decreases, the OS 32 generates interrupts to decrease CPU speed.” *Id.* The “CPU utilization” teachings disclose sending a request to change a clock frequency in response to a predefined change in performance of the first master device (a predefined

change in utilization of the CPU). Ex. 1002, ¶¶235, 238-242.

If Patent Owner argues that Shaffer's interrupt request is not sent by a first master device (the CPU 20), but rather by the operating system (OS 32), it would have been obvious that the OS would have executed on a master device, such as CPU 20 (a master device per the '759 patent), as discussed in §XI.A.1(b).

Shaffer discloses or renders obvious this element. Ex. 1002, ¶¶235-237, 241-242.

(d) “wherein the predefined change in performance is due to loading of the first master device as measured within a predefined time interval”

Shaffer in view of Lint renders this limitation obvious. As explained, §XI.A.1(c), Shaffer teaches requesting a clock frequency change based on a predefined change in performance of the CPU 20. That predefined change is due to the loading of the first master device (the CPU utilization percentage). Ex. 1005, 4:50-5:30; Ex. 1002, ¶¶235, 238-240.

Shaffer sets a lower clock speed based on monitoring during a predefined time interval. Ex. 1005, 4:6-8 (“[T]he CPU 20 operates at a lower speed when the OS 32 determines that no processing is occurring or has not occurred for a predetermined amount of time.”), 5:48-52 (“Processing performance can be calculated in a variety of ways. In particular, the requirement may be a time requirement such as five microseconds, indicating that the task needs service every five microseconds.”); Ex. 1002, ¶¶238-240.

If Patent Owner argues that the predefined time interval discussion in Shaffer is not applicable to changing the frequency of a high-speed clock frequency signal, Lint also discloses this limitation. Ex. 1006, 3:1-7 (changing the “performance state ... based in part on the data representing the average performance over the previous period of time.”). This “average performance” in Lint indicates loading percentage of the processor as measured within a predefined time interval. *Id.*, 3:15-19 (“[A]verage performance provides information back to the OS indicating

the percentage number of cycles spent doing useful work (not idle) in the last predetermined interval...”). The OS then determines the P-state, which determines the clock speed and voltage, for the next interval. Ex. 1006, 3:8-11, 1:18-67 (explaining that a P-state is a frequency and voltage operating point). Ex. 1002, ¶¶83, 222, 243-247.

Implementing Lint’s feature in Shaffer’s OS CPU monitor would monitor the CPU for a predefined change in performance due to the loading (“percentage number of cycles doing useful work”) of the first master device (CPU 20 executing the modified OS 32) as measured within a predefined time interval (“predetermined interval”). Ex. 1002, ¶¶222-223.

It would have been obvious to a POSA to modify Shaffer’s CPU monitor to use Lint’s “predetermined interval.” Shaffer describes a “CPU utilization percentage,” without explaining how that percentage would have been calculated. Lint provides more detail regarding measuring a utilization percentage—as a “percentage number of cycles spent doing useful work (e.g., not idle) in the last predetermined interval.” Ex. 1006, 3:6-18. Lint’s detailed a well-known technique that easily could have been used in Shaffer’s system. *See KSR Int’l Co. v. Teleflex Inc.*, 550 U.S. 398, 420-21; *AstraZeneca LP v. Breath Ltd.*, 603 F. App’x 999, 1002. Ex. 1002, ¶¶222-223.

It would have been obvious to measure Shaffer’s changes in CPU performance as Lint described. Shaffer explains that prior art systems adjusted clock speeds based on “the amount and frequency of system activity.” Ex. 1005, 1:59-63. Shaffer explains that CPU utilization measurements may be performed by “CPU monitoring programs” that were “widely available and, in many instances, are included with the operating system software.” *Id.*, 4:54-56. Such programs were well-known at the time of Shaffer. A 1991 textbook, titled THE ART OF COMPUTER SYSTEMS PERFORMANCE ANALYSIS, features a “Monitors” chapter describing monitors that “observe the performance of systems” and “measure resource utilizations ... to find the performance bottleneck.” Ex. 1009, 93. Lint’s CPU utilization monitoring constitutes a natural

progression these well-known performance monitoring systems. A POSA would have appreciated Lint as a type of monitor. Lint recognizes pre-existing CPU utilization monitoring technology: “...current operating systems use this technique so that the processor is placed at an operating frequency that matches the processor utilization.” Ex. 1006, 1:31-34, Figure 1 (showing “Prior Art” that “re-compute[s] a new P state from the busy ratio and current performance”). Ex. 1002, ¶¶208-214, 222-226.

It would have been conventional and routine to combine Lint’s CPU utilization monitoring over a predetermined time interval with Shaffer’s CPU utilization monitor. Both Shaffer and Lint disclose multi-processor systems with an OS that monitors CPU utilization to control processor operating frequency. Ex. 1005, 4:51-54 (“[T]he OS 32 is used to control the frequency of the clock module 50 in response to a CPU utilization application that dynamically monitors the level of CPU usage”), Claim 1 (“one or more CPUs”); Ex. 1006, 1:28-34 (“Some current operating systems use this technique so that the processor is placed at an operating frequency that matches the processor utilization.”). Ex. 1002, ¶¶216-226.

Both Shaffer and Lint address selecting clock frequency to save power. Ex. 1005, 1:38-40 (“Power saving features have become increasingly important over the past few years because of the dramatic increase in clock speeds.”), 1:49-51; Ex. 1006, 1:18-25 (“Mobile computers ... typically incorporate certain power management techniques. One such technique ... provides for changing both the operating voltage and frequency of the processor such that transition to a lower frequency (e.g., performance) point leads to a cubic reduction in power consumption ... at the lower frequency point.”). Ex. 1002, ¶216.

Shaffer refers to such “CPU monitoring programs” as “widely available.” Ex. 1005, 4:54-55. Lint acknowledges available programs. Ex. 1006, 1:31-34. Lint teaches improvements to the technique, which receives “data from a processor representing an average performance of a

processor ... over a previous period of time and determin[es] a performance state (P-state) for a next period of time based in part on the data representing the average performance over the previous period of time.” Ex. 1006, 3:2-7. A POSA would have been motivated to employ Lint’s CPU utilization monitoring technique in Shaffer’s OS to better interface with processor chips featuring hardware coordination of P-states, to benefit by “provid[ing] for the best performance while saving power.” Ex. 1006, 2:31-33. Ex. 1002, ¶¶208-214, 222-226.

Using Lint’s teachings in Shaffer’s system would have involved nothing more than use of a known technique (measuring average utilization of a CPU over a predetermined time interval) to improve similar devices (Shaffer’s similar system that measures CPU utilization percentages) in the same way (to “provide for the best performance while saving power.”). Ex. 1002, ¶¶214-221.

- (e) **“in response to receiving the request from the first master device: providing the clock frequency of the high-speed clock as an output to control a clock frequency of a second master device coupled to the bus”**

Shaffer discloses in response to receiving the request (instruction from OS 32 along line 19 or from CPU along line 49) from the first master device (CPU 20 executing OS 32): providing the clock frequency of the high-speed clock as an output (the signal from the clock module 50 along lines 23, 25) to control a clock frequency of a second master device (memory controller 22, bus controller 24, and/or another CPU) coupled to the bus (bus 21). Ex. 1002, ¶¶231-233, 248.

Shaffer explains, “[N]ot only the CPU, but often the entire system bus operates off of a single clock, so all of the chip sets including I/O controllers, memory, and other components draw higher currents at higher clock speeds than sometimes necessary for efficien[cy].” Ex. 1005, 1:46-51. Likewise, “the CPU speed control system 18 is adapted to provide a programmably variable clock frequency to the other controllers and buses in the system, such as the memory controller 22 and/or the system or peripheral bus controller 24.” *Id.*, 4:15-21. “[L]owering of the clock frequency...,” as Shaffer states, “further contributes to overall power savings.” *Id.*, 4:21-23. Ex. 1002, ¶248.

Shaffer's controllers 22, 24 are examples of the '759 patent's second master device(s). Ex. 1001, 3:22-23 (“[M]aster device ... can be ... an input/output bus controller”). “[T]he clock module 50 supplies a clocking signal to the memory controller 22 through a memory clock control line 23 and supplies a clocking signal to the system bus controller 24 through a system bus clock control line 25.” Ex. 1005, 4:26-29. Shaffer describes, “when the OS 32 sends a signal to the clock module 50 instructing it to provide the CPU 20 with the clock speed [] associated with an OS_idle signal, both the memory controller 22 and the system controller 24 operate at the specified lower clock speed. Similarly, once the system leaves the OS_idle state and the clock output of the clock module 50 increases, the memory controller 22 and the controller 24 likewise speed up.” *Id.*, 4:40-50.

Shaffer teaches this limitation. Ex. 1002, ¶¶229-233, 248.

(f) “and, providing the clock frequency of the high-speed clock as an output to control a clock frequency of the bus”

Shaffer's clock module 50 provides a clock signal not only to the CPU, but also to the system bus(es). Ex. 1005, 2:17-19 (“The system includes a programmable frequency synthesizer for providing the CPU and other system buses in the device with a variable clocking frequency.”), Abstract; Ex. 1002, ¶248. Shaffer outputs this clock signal to both the bus and other devices. Ex. 1005, 4:15-25 (“[T]he CPU speed control system 18 is adapted to provide a programmably variable clock frequency to the other controllers and buses in the system.”). Shaffer teaches, “the clock module 50 drives the entire system bus [] and thereby reduces power requirements for the processor, related chipsets, memory, controllers and the like.” *Id.*, 5:66-6:2.

Shaffer in view of Lint renders claim 1 obvious. Ex. 1002, ¶¶248, 251.

2. Claim 14

(a) “A system comprising: a bus capable of operation at a variable clock frequency”

As described, §XI.A.1(f), Shaffer teaches “[t]he system includes a programmable frequency

synthesizer for providing the CPU and other system buses in the device with a variable clocking frequency.” Ex. 1005, Abstract, 2:17-19. Ex. 1002, ¶¶269-270.

- (b) “a first master device coupled to the bus, the first master device configured to provide a request to change a clock frequency of a high-speed clock in response to a predefined change in performance of the first master device,”

As described, §§XI.A.1(a) to XI.A.1(c), Shaffer discloses this.

- (c) “wherein the predefined change in performance is due to loading of the first master device as measured within a predefined time interval; and”

As described, §XI.A.1(d), Shaffer in view of Lint renders obvious this limitation.

- (d) “a programmable clock controller having an embedded computer program therein, the computer program including instructions to:”

Shaffer teaches a programmable clock controller. Ex. 1005, 3:8-11 (“The exemplary CPU control system 18 includes a CPU 20 coupled to an intelligent programmable clock module 50 having a frequency synthesizer for providing the CPU 20 with a clocking signal.”). Shaffer indicates that the clock frequency provided by the clock module 50 is “programmably variable.” *Id.*, 2:65, Claim 6. Ex. 1002, ¶284.

A POSA would have understood that references to “intelligent,” “programmable,” and “programmably,” regarding Shaffer’s clock module 50, referred to a computer program embedded in the clock control module. It would have further been obvious that where Shaffer describes clock module performing functionality, the clock module would have included instructions that perform the functionality referenced in §§XI.A.2(e)–XI.A.2(g). Ex. 1002, ¶¶28-30, 284-285.

- (e) “receive the request provided by the first master device;”

As described, §XI.A.1(b), Shaffer discloses this. Ex. 1005, 8:23-26 (“said intelligent clock module adapted to receive one or more signals from an OS to enable varying of said operating frequency.”). Ex. 1002, ¶286.

- (f) **“provide the clock frequency of the high-speed clock as an output to control a clock frequency of a second master device coupled to the bus in response to receiving the request provided by the first master device; and”**

As described, §XI.A.1(e), Shaffer teaches this.

- (g) **“provide the clock frequency of the high-speed clock as an output to control the variable clock frequency of the bus in response to receiving the request provided by the first master device.”**

As described, §XI.A.1(f), Shaffer teaches this.

Shaffer in view of Lint renders claim 14 obvious. Ex. 1002, ¶¶198-226, 269-290.

3. Claim 17

- (a) **“The system of claim 14,”**

As described, §XI.A.2, Shaffer in view of Lint renders claim 14 obvious.

- (b) **“wherein the instructions to provide the clock frequency of the high-speed clock as an output to control the variable clock frequency of the bus include instructions to adjust the clock frequency of the bus.”**

As described, §XI.A.1(f), Shaffer teaches that when the clock module 50 raises or lowers the clock frequency, this new clock frequency signal is provided to both the other devices and the bus, which controls and adjusts the clock frequency of the bus. Ex. 1005, 2:17-19 (“The system includes ... providing the CPU and other system buses in the device with a variable clocking frequency.”), Abstract. Shaffer teaches, “[T]he CPU speed control system 18 is adapted to provide a programmably variable clock frequency to the other controllers and buses in the system.” *Id.*, 4:15-18. Shaffer teaches, “[T]he clock module 50 drives the entire system bus.” *Id.*, 5:66-6:2. Ex. 1002, ¶¶260-261.

Shaffer in view of Lint renders claim 17 obvious. Ex. 1002, ¶¶260-261, 291-293.

B. Ground II: Claims 18, 21-22, 24 are Obvious Over Shaffer in View of Lint and Kiriake

1. Claim 18

- (a) “A system comprising: a bus capable of operation at a variable clock frequency”

As described, §XI.A.1(f), Shaffer teaches “[t]he system includes a programmable frequency synthesizer for providing the CPU and other system buses in the device with a variable clocking frequency.” Ex. 1005, Abstract, 2:17-19. Ex. 1046, ¶¶344, 378.

- (b) “a first master device coupled to the bus”

As described, §§XI.A.1(a) to XI.A.1(c), Shaffer discloses this. Ex. 1046, ¶¶378-383.

- (c) “an arbiter coupled to the bus and coupled to the first master device, the arbiter configured to control flow of data on the bus; and”

Kiriake teaches master devices (“Peripheral Processor 1” 10, “Peripheral Processor 2” 11) coupled to system bus 12, and arbiter 2 “that performs arbitration relating to use of this system bus.” Ex. 1028, ¶¶0011, 0022; Fig. 1 (annotated below). Ex. 1046, ¶206.

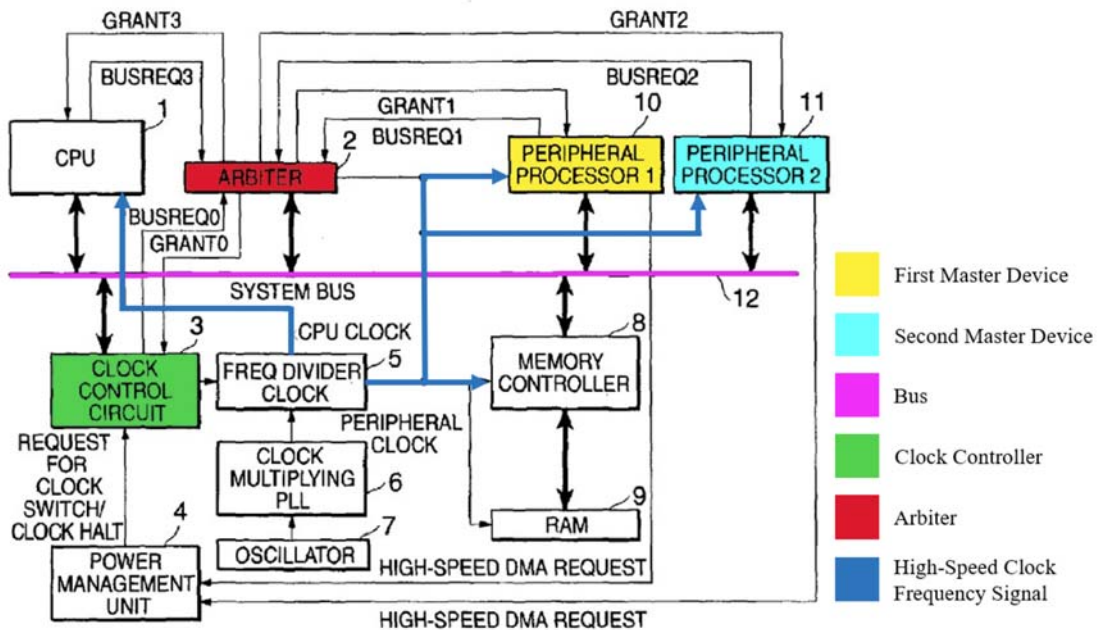


FIG. 1

Kiriake's "arbiter 2 arbitrates the bus request made through the bus request signal (BUSREQ). The arbiter 2 then asserts the bus grant signal (GRANT) to the CPU 1 or peripheral processor 10 or 11 that is recognized as the bus master as a result of arbitration." Ex. 1028, ¶0027. Ex. 1046, ¶¶207-212.

Kiriake teaches this limitation. Ex. 1046, ¶¶386-392.

It would have been obvious to modify Shaffer's system to include Kiriake's arbiter. A POSA would have been motivated to do so with a reasonable expectation of success for several reasons. Ex. 1046, ¶¶365-374.

Shaffer discloses, "[S]ystem bus controller 24 may be coupled to optional peripheral devices." Ex. 1005, 4:38-39. Shaffer teaches that "variable clocking frequency" is provided to both the CPU and "other system buses in the device." Ex. 1005, Abstract. A POSA would have understood Shaffer's system bus 21 to be a limited data transfer resource shared among these multiple devices (CPU 20 and "peripheral devices"), and that to implement the "programmably variable clock frequency" among these bus devices to obtain the advantages of such variable clock control described in Shaffer, their access to, and frequency of operation on, system bus 21 would be arbitrated. Ex. 1046, ¶¶369-374.

Kiriake teaches that arbiter 2 arbitrates access to, and frequency of operation on, system bus 12 for CPU 1 and peripheral processors 10, 11 (Shaffer's "peripheral devices"), to control the flow of data transfer, e.g., "between memory and peripheral equipment." Ex. 1028, ¶0006. A POSA would have been motivated to combine Kiriake's arbiter 2 with Shaffer's system 18 to control bus transactions with peripheral devices. Ex. 1046, ¶¶369-374.

It was well-known to add a bus arbiter to a bus controller. Ex. 1042, 6:64-7:10, Fig. 3. Adding Kiriake's arbiter to Shaffer's system would have involved nothing more than combining prior art elements (Kiriake's arbiter 2 with Shaffer's frequency control system 18) according to

(d) “a clock controller coupled to the arbiter and coupled to the first master device,”

[illegible]

(e) “the clock controller configured to output a clock frequency of a high-speed clock to control the variable clock frequency of the bus and to control a clock frequency of a second master device coupled to the bus,”

(f) “the clock controller configured to receive a request to change the clock frequency of the high-speed clock from the first master device,”

- 36 -

operating frequency.”). Ex. 1046, ¶410.

- (g) “the request sent from the first master device in response to a predefined change in performance of the first master device”**

As described, §§XI.A.1(a) to XI.A.1(c) Shaffer discloses this. Ex. 1046, ¶410.

- (h) “wherein the clock controller is configured to adjust the variable clock frequency of the bus in response to receiving the request from the first master device, and”**

As described, §§XI.A.1(e) and XI.A.1(f), Shaffer teaches this. Ex. 1046, ¶410.

- (i) “wherein the predefined change in the performance is due to loading of the first master device as measured within a predefined time interval.”**

As described, §XI.D.1(d), Shaffer in view of Lint renders obvious this limitation. For the reasons described, a POSA would have combined Lint with Shaffer and Kiriake, and would have had a reasonable expectation of success at least because they describe similar systems in the same field.

Shaffer in view of Lint and Kiriake renders claim 18 obvious. Ex. 1046, ¶410.

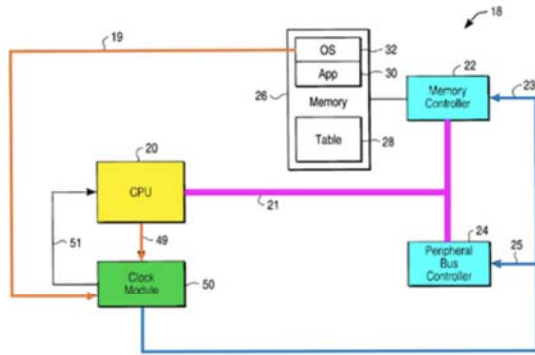


Figure 1 of Shaffer

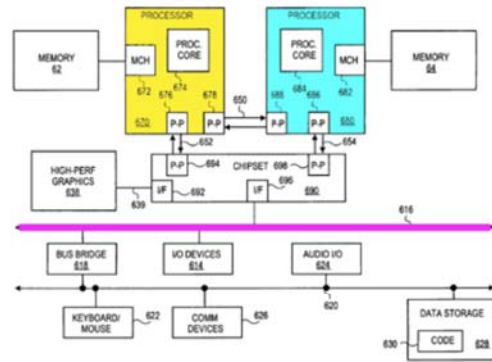


Figure 6 of Lint

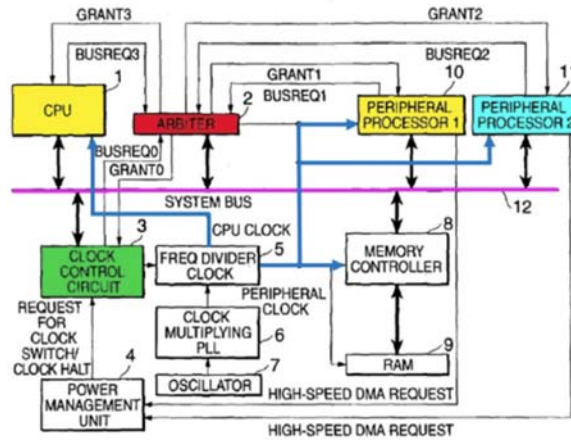


Figure 1 of Kiriak

2. Claim 21

(a) “The system of claim 18, wherein”

See §XI.B.1.

(b) “adjusting the variable clock frequency of the bus comprises decreasing the clock frequency of the bus.”

Shaffer teaches, “[T]he CPU [] can instruct through line 49 the clock module 50 to increase or decrease the output frequency...” with the decreased frequency signal then provided to “other controllers and buses.” Ex. 1005, 3:8-22, 4:15-18, 5:66-6:2. Ex. 1046, ¶418.

3. Claim 22

(a) “The system of claim 18, wherein”

See §XI.B.1.

- (b) **“adjusting the variable clock frequency of the bus comprises selecting the variable clock frequency to be a frequency divisible by a factor of 1, 2, 4, 8, or 16.”**

Shaffer teaches selecting the variable clock frequency of the CPU and the bus to be “40 MHz” (40,000,000 Hz) “for a particular program,” Ex. 1005, 5:12, which is divisible by 16, etc. Ex. 1046, ¶¶423-424.

4. **Claim 24**

- (a) **“The system of claim 18, wherein”**

See §XI.B.1.

- (b) **“the predefined change in the performance of the first master device comprises a variation in load of the first master device.”**

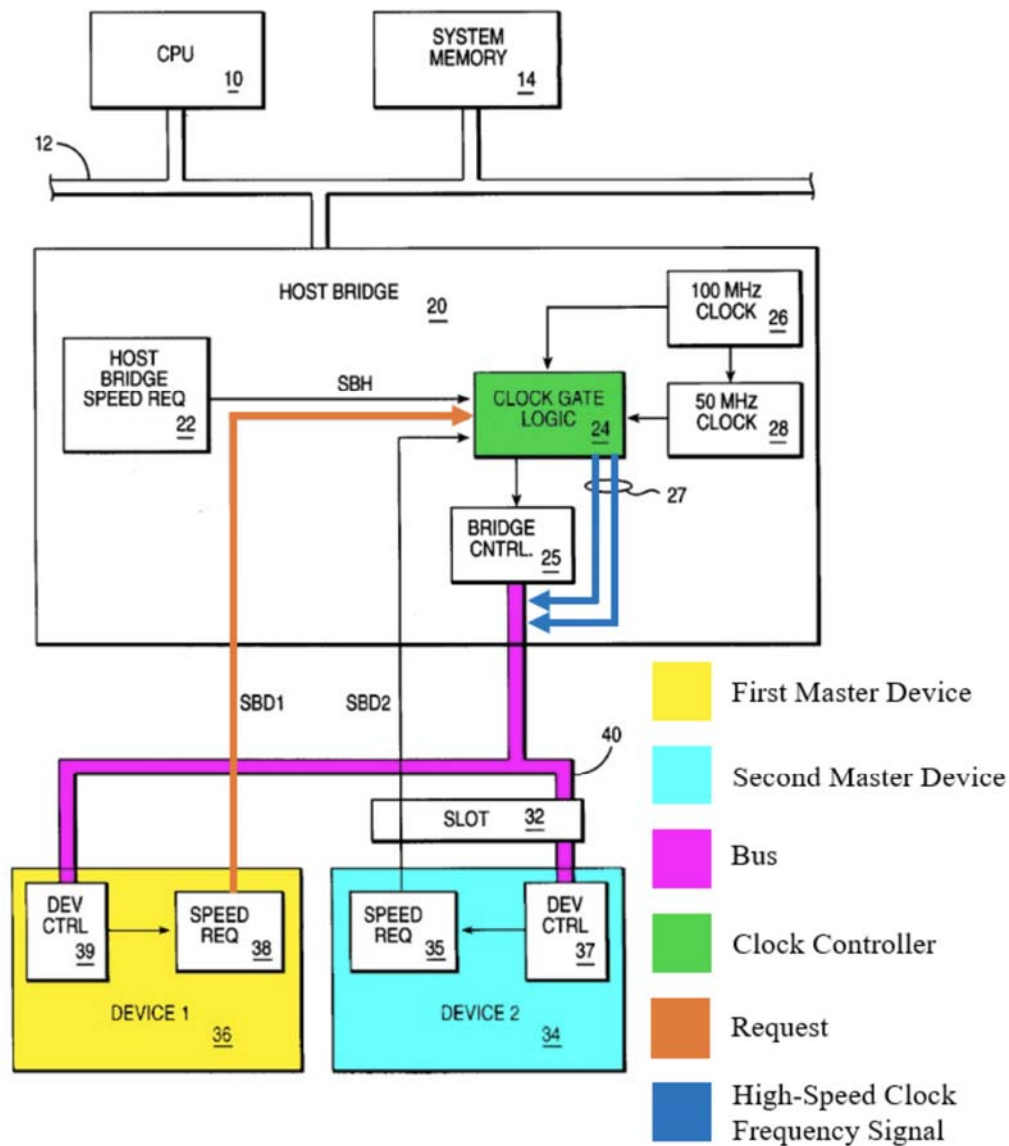
As described, §§XI.B.1(c) and XI.B.1(d), Shaffer teaches this. Ex. 1005, 4:50-5:30. Ex. 1046, ¶¶331-343, 430-434.

C. Ground III: Claims 1, 14, 17 are Obvious Over Chen in View of Terrell

1. Claim 1

(a) “A method comprising: monitoring a plurality of master devices coupled to a bus”

Chen discloses a plurality of devices (“Device 1” 36 and “Device 2” 34) coupled to a bus (I/O bus 40). Ex. 1003, Fig. 1, 2:67-3:3. Ex. 1002, ¶147.



Devices 34, 36 are master devices. Ex. 1003, 4:42-43 (“If any of the devices [34, 36] are either the controlling master...”), 3:61-64 (“If a selected target (and the requesting master) can both

handle 100Mhz data transfers....”). The ’759 patent explains that master devices may “initiate communication.” Ex. 1001, 3:11-21. Chen teaches that devices 34, 36 may initiate communications. Ex. 1003, 4:48-51 (“if device 36 initiates a read operation to obtain data”); 6:42-44 (“[I]t is assumed that device 36 initiates a read operation...”). Ex. 1002, ¶147.

Chen’s devices 34, 36 each generate a signal (“SBD1 for device 36 or SBD2 for device 34”). Ex. 1003, 3:61-64, Fig. 1. Chen’s clock gate logic 24 monitors these signals. Ex. 1003, 3:56-61 (“As noted above, clock gate logic 24 in bridge 20 receives several side band signals (SBH, SBD1, SBD2) which are used by host bridge 20 and the various I/O devices 34 and 36 to indicate the data phase speed which they can operate.”). Ex. 1002, ¶¶148-149.

(b) “receiving a request, from a first master device of the plurality of master devices, to change a clock frequency of a high-speed clock”

Chen teaches receiving a request (signal SBD of speed request circuit 38), from a first master device of the plurality of master devices (device 36) to change a clock frequency of a high-speed clock (clock line 27 output by clock gate logic 24). Ex. 1003, 3:61-64. Ex. 1002, ¶150.

“Device control logic 37, 39” in each device 34, 36 “determines the speed at which devices 34 and 36 can run for a particular transaction.” Ex. 1003, 3:5-7. Devices 34, 36 could operate at 50Mhz, and the device control logic 37, 39 could determine that “higher frequency operation” at 100Mhz “is desired.” *Id.*, 3:11-12. Based on determining transaction type, device control logic 37 will “transmit a control signal to speed request circuit 35.” *Id.*, 3:9-12. The speed request circuit 35 will “output a control signal to clock gate logic circuit 24.” *Id.*, 3:14-15. “The control signals from device 36 and 34 are also side band signals shown as SBD1 and SBD2, respectively (for devices 1 and 2).” *Id.*, 3:31-33. If devices 34, 36 are running at 50Mhz, and device control logic 37, 39 determines 100Mhz clock speed for a transaction, then “they will drive their respective side band signal to logical zero (ground) during the address phase of the transaction.” *Id.*, 4:44-46, 4:22-28.

After receiving these signals, “[c]lock gate circuit 24 causes the frequency of bus 40 to be dynamically changed (gated) by transmitting the appropriate device unique clock lines 27.” *Id.*, 3:20-22. Ex. 1002, ¶¶149-151.

Chen therefore discloses the clock gate logic circuit 24 receiving a request (side band signal SBD1) from a first master device (36) of the plurality of master devices (34, 36), to change a clock frequency of a high-speed clock (the high-speed clock frequency signal of clock lines 27).³ When the devices 34, 36 and bus 40 are running at lower speed (50Mhz), setting the side band signals SBD1 and SBD2 to logical zero (ground) is therefore a “request to change” the high-speed clock frequency signal of clock lines 27 of clock gate logic 24 (from 50Mhz to 100Mhz). *Id.*, 4:44-46, 4:22-28. Such requests also “identify an appropriate frequency” (e.g., 100Mhz when grounded), which the applicant argued (during prosecution) is required of a master device’s request, to change a clock frequency of a high-speed clock. See Ex. 1013, 9; Ex. 1003, 4:26-28, 3:65-67.

Chen teaches this limitation. Ex. 1002, ¶¶149-151, n. 6.

(c) “the request sent from the first master device in response to a predefined change in performance of the first master device”

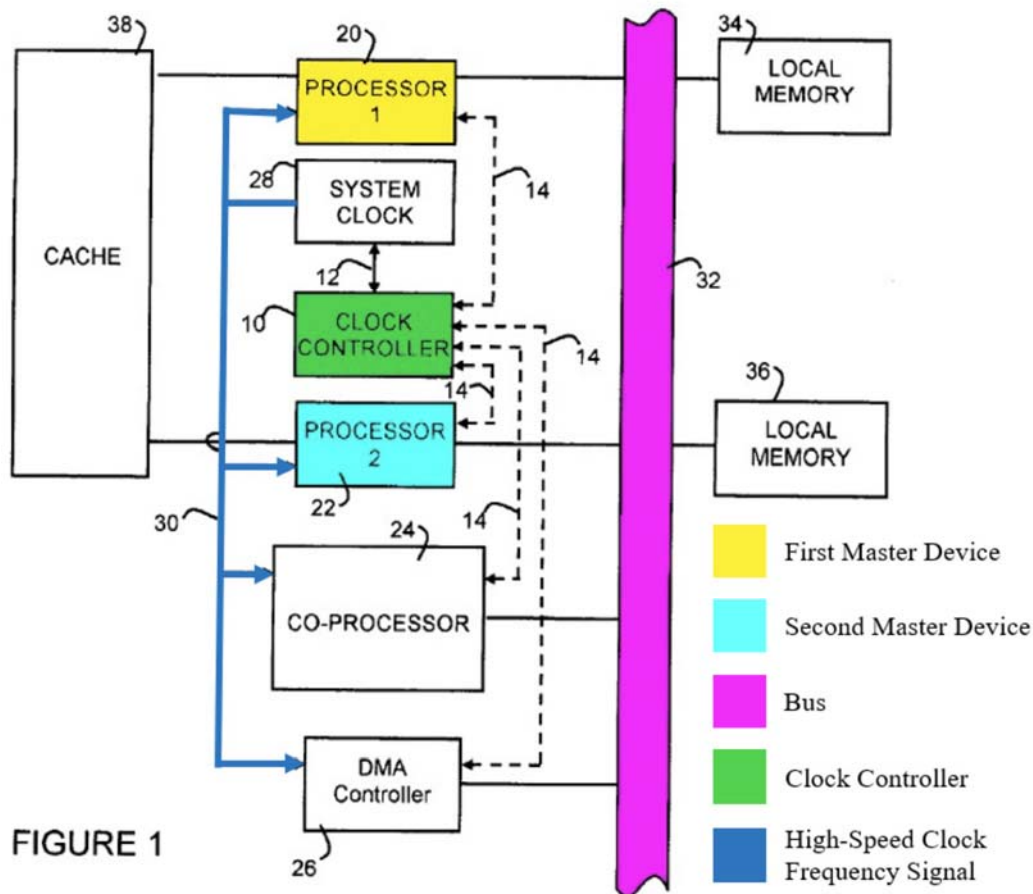
Chen teaches that the request (signal SBD) sent from the first master device (device 36) is in response to a determination of “the speed at which device[36] can run for a particular transaction.” Ex. 1003, 3:6-7. The “device control logic” on devices 34 or 36 “will determine what type of transaction is being run and transmit a control signal to speed request circuit 35 [or 38] when the higher frequency operation is desired.” *Id.*, 3:5-13. Ex. 1002, ¶¶150, 153.

A predefined change in performance may be “a variation in the load or the output of the device,” or more generally, “an event such as a desired increase in device performance.” Ex. 1001,

³ Chen describes “a high frequency clock and a low frequency clock,” of 100Mhz and 50MHz, respectively. The ’759 patent describes that “clock frequency of the high-speed clock” includes clock controller signals as low as 1Mhz. Ex. 1001, 3:52-60. Thus, both clock lines 27 from Chen’s clock gate logic 24 are “high-speed” clock signals.

3:64-4:19. Chen discloses determining a “particular transaction” (event) where “higher frequency operation is desired.” Ex. 1003, 3:5-13. Ex. 1002, ¶¶152-153.

If Patent Owner argues that Chen’s “type of transaction” determination is not a “predefined change in performance of the first master device,” Terrell also teaches this limitation. Terrell discloses adjusting a system clock frequency based on predefined changes in performance of master devices (processors 20, 22, or processing elements 40, 42). Ex. 1002, ¶154.



Terrell teaches “[t]here are two basic steps employed in the present invention:

1. Over a sample period, measure how many clock cycles are being used by each processing element that is attached to the shared clock.
2. Adjust the system clock frequency to provide the minimum number of clock cycles required by the processing element that is using the largest number of clock

cycles.”

Ex. 1004, ¶¶0025-0027, 0035 (describing clock usage counters that measure the number of clock cycles that do and do not perform work), Fig. 2. “If the busiest processor is idle half of the time ... then the system clock frequency can be lowered. ... If the busiest processor is not idle enough over a sample period, the clock frequency is increased ...” *Id.*, ¶¶0050, 0052, 0054 (“[T]he method and apparatus ... is capable of varying the system clock frequency both upwardly and downwardly, as a function of the measured ‘idleness’ of the processors.”). Ex. 1002, ¶¶154-156. It would have been obvious to modify Chen’s master devices that already send requests “to change a clock frequency of a high-speed clock,” to send those requests “in response to a predefined change in performance of the master devices” (based on “how many clock cycles are being used by each processing element”), as taught by Terrell, and a POSA would have had several motivations to do so. Ex. 1002, ¶¶126-142, 145.

First, it was well-known, desirable, and taught by Terrell to save power. See Ex. 1004, ¶0005. Reducing clock speed was a well-known technique for reducing power consumption. Ex. 1002, ¶138.

Chen teaches embodiments for a cost-oriented solution and/or low-frequency operations for saving power. Ex. 1003, 5:21-24 (“The remaining devices on the bus, [] not involved in the current high frequency data transfer transaction, will continue to receive the lower frequency clock.”), 4:36-39 (“[One can use] the high frequency mode of operation in other transactions, if the system needs require high frequency and additional cost and complexity is not a factor.”), 3:25-29, 3:42-44. Therefore, a POSA working with Chen’s system would have known that operating at a higher than needed clock speed would have increased costs by using excess power; the POSA would have been motivated to seek ways to reduce clock frequency without negatively impacting the performance, as Terrell teaches. Ex. 1002, ¶¶128-129.

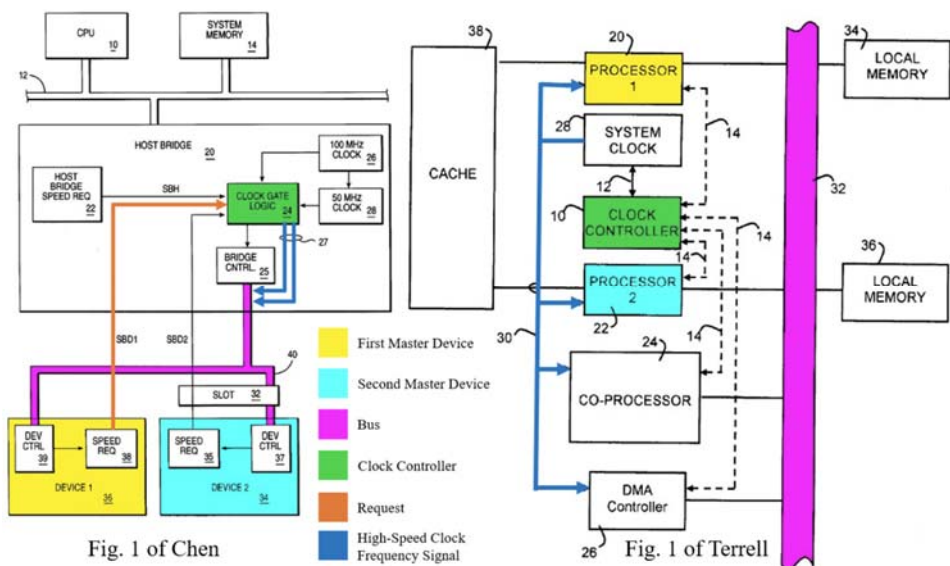
Terrell explains, “it is desirable to [] reduce the frequency of a shared clock to the minimum frequency that allows the processing elements to function correctly while using the least amount of power.” Ex. 1004, ¶0005. Terrell teaches need-based heuristics for measuring the master devices’ performance needs and selecting the clock speed “to match [the] minimum processing requirements.” *Id.*, ¶0048. Terrell teaches performance measurement of the master devices by “software implementation” that “can be used for measuring clock cycle consumption by the processor elements.” *Id.*, ¶0037. Terrell’s software-based technique “can be embodied in a computer program that is run in each of the processors, and with the measured result being passed to a controller functional block such as clock controller 10 [], for adjustment of the system clock.” *Id.*, ¶0041. Ex. 1002, ¶¶130-138.

Chen teaches capability-based heuristics for selecting the highest clock speed at which the devices can operate. Ex. 1003, 6:55-58 (“[T]he present invention is capable of dynamically changing the frequency of an I/O bus based on the capability of the devices connected thereto.”), 4:40-46, 5:56-65. A POSA would have been motivated to modify Chen’s capability-based heuristic with Terrell’s need-based heuristic to select a clock frequency that increases the devices’ frequency only when needed, to reduce power consumption, even if the devices can use higher speeds. Ex. 1002, ¶¶132-137.

A POSA would have been motivated to use Terrell’s clock speed control solution as software in Chen’s microcontrollers to control the master devices 34, 36 based on the clock speed needed for the transaction rather than the device’s maximum clock speed. Ex. 1002, ¶137. Adjusting the speed of Chen’s master devices during a transaction or series of transactions would thus “minimize the cost and complexity of providing high frequency operation.” Ex. 1003, 4:32-33. To do so as Terrell describes would permit Chen’s system “to function correctly while using the least amount of power,” Ex. 1004, ¶0005. Ex. 1002, ¶136.

Second, Terrell states that its control features are “desirable” to include in bus systems like Chen’s systems. Ex. 1004, ¶¶0006, 0008 (“Examples of applications where such a control feature would be desirable include ... An on-chip bus that hosts two or more bus masters, all of which share a common bus clock.”). As discussed, §XI.C.1(a), Chen describes an on-chip bus that hosts two or more bus masters all sharing a common bus clock. Terrell’s express teaching would have motivated a POSA to add Terrell’s need-based clock speed request determination to Chen’s master devices (for example, in the control logic 37, 39). Chen describes the control logic as the logic that makes capability-based clock speed requests. *See* Ex. 1003, 3:3-19, 4:42-46. “[T]he question is whether there is something in the prior art as a whole to suggest the *desirability*, and thus the obviousness, of making the combination.” *In re Fulton*, 391 F.3d 1195, 1200 (Fed. Cir. 2004) (cleaned up). Ex. 1002, ¶138.

Third, it would have been obvious to combine Chen with Terrell in the *above* manner because this would have involved nothing more than use of a known technique (measuring master devices’ utilization and using that measurement to request clock speed changes) to improve a similar device (Chen’s clock speed control system) in the same way (using Terrell’s frequency control logic with Chen’s clock speed control system). Ex. 1002, ¶¶139-140.



A POSA would have known that using Terrell’s need-based clock speed request

determination software in Chen's device control logic would have involved nothing more than combining prior art elements (Terrell's clock cycle usage monitoring software and Chen's device control logic 37, 39) according to known methods (configuring Chen's device control logic 37, 39 to execute Terrell's clock cycle usage monitoring software in making clock request determinations) to yield predictable results (a master device that makes clock request determinations based on clock cycle usage monitoring).

Accordingly, Chen in view of Terrell renders this limitation obvious. Ex. 1002, ¶¶131-137, 157.

(d) “wherein the predefined change in performance is due to loading of the first master device as measured within a predefined time interval”

As discussed, §XI.C.1(c), Chen and Terrell teach a master device requesting a clock speed change “in response to a predefined change in performance of the first master device.” Terrell teaches that the predefined performance change is due to loading (workload) of the first master device (processor 20) as measured within a predefined time interval (a “sample period”). Ex. 1002, ¶¶154-156.

Terrell teaches clock usage counters (46, 48) that measure “the number of clock cycles that are used by each Processing Element to perform work, as well as the number of clock cycles that are not used.” Ex. 1004, ¶0035. Terrell measures the processing element's workload over a “sample period.” *Id.*, ¶0042. This is one of Terrell's “two basic steps.” *Id.*, ¶¶0025-0026. This measurement reflects the “loading of the first master device” (the processor's workload) “as measured within a predefined time interval” (a sample period). Ex. 1002, ¶154-156. Terrell teaches, “[t]his information is then passed along to the clock controller ... for use in adjusting the system clock 28.” Ex. 1004, ¶0044. “[T]he system clock frequency” is adjusted to provide “enough clock cycles to process the current system workload.” *Id.*, ¶0053. Ex. 1002, ¶156.

As discussed, §XI.C.1(c), a POSA would have found it obvious to, and would have been motivated to, combine Terrell's performance measurement technique with Chen's device control logic that sends requests to change the clock frequency. Ex. 1002, ¶¶122-142, 145.

Chen in view of Terrell renders this limitation obvious. Ex. 1002, ¶162.

- (e) **“in response to receiving the request from the first master device: providing the clock frequency of the high-speed clock as an output to control a clock frequency of a second master device coupled to the bus”**

Chen discloses: in response to receiving the request (control signal SBD) from the first master device (device 36), providing the clock frequency of the high-speed clock (clock line 27 output by clock gate logic 24) as an output to control a clock frequency of a second master device (device 34) coupled to the bus (I/O bus 40). Ex. 1002, ¶157.

Chen teaches that devices exchanging data receive the same clock frequency. Ex. 1003, 5:13-24 (“[T]he controlling bus master device and the selected target which both indicate they are capable of operating at 100Mhz data phases will be clocked at 100Mhz.”). Ex. 1002, ¶¶157-159.

As discussed, §XI.C.1(a), Chen teaches that master devices 34, 36 may be coupled to bus 40 and participate in “peer to peer transactions.” Ex. 1003, 5:59-65. Both master devices 34, 36 would be provided with the same clock frequency of the high-speed clock (signal of clock line 27) as an output (from clock gate logic 24): “if device 36 requires data from device 34, and both of these devices activate their side band signals (SBD1 and SBD2, respectively), then the data can be transferred at the higher frequency (100Mhz, if both devices are enabled to operate at that speed).” Ex. 1003, 5:61-65. Ex. 1002, ¶¶159-161.

Chen teaches that, during high frequency transactions, the high-speed clock's clock frequency may be outputted to control the clock frequency of a first and a second master device coupled to a bus.

Chen teaches this limitation. Ex. 1002, ¶¶159-161.

(f) “providing the clock frequency of the high-speed clock as an output to control a clock frequency of the bus”

Chen teaches providing the clock frequency of the high-speed clock (clock line 27 output by clock gate logic 24) as an output to control a clock frequency of the bus (bus 40). Chen teaches that when “[i]n response to” a frequency control signal, “control logic in the bridge chip causes the higher frequency clock in the bridge chip to be activated such that the host bridge, bus and I/O device are all then operating at the higher frequency.” Ex. 1003, 2:8-14. For example, “[c]lock gate logic 24 will then enable the high frequency clock 26 and drive bus 40 at 100Mhz.” Ex. 1003, 4:63-5:5, 5:59-65, Claim 13. Ex. 1002, ¶157.

Claim 1 is therefore invalid under 35 U.S.C. §103 as being obvious over Chen in view of Terrell. Ex. 1002, ¶162.

2. Claim 14

(a) “A system comprising: a bus capable of operation at a variable clock frequency”

As explained, §§XI.C.1(b), XI.C.1(f), and XI.C.2, Chen controls bus 40 (and devices 34, 36) at clock frequencies of 50 MHz or 100 MHz. Chen changes the bus clock frequency signal between 50 MHz and 100 MHz, for example, depending on transaction type performed. Ex. 1003, 4:6-10 (“Data at either the lower (e.g. 50 Mhz) or higher (e.g. 100 Mhz) transfer rate would be clocked onto the bus on the rising edge of the clock signal...”), 4:1-6. Chen expressly describes changing the frequency of bus 40 “dynamically.” *Id.*, 3:20-21. Chen’s bus 40 can operate at a variable clock frequency between 50Mhz and 100Mhz. *Id.*, 4:6-10. Ex. 1002, ¶182.

(b) “a first master device coupled to the bus, the first master device configured to provide a request to change a clock frequency of a high-speed clock in response to a predefined change in performance of the first master device,”

As described, §§XI.C.1(a) through XI.C.1(c), Chen in view of Terrell renders obvious this limitation.

- (c) **“wherein the predefined change in performance is due to loading of the first master device as measured within a predefined time interval; and”**

As described, §XI.C.1(d), Chen in view of Terrell renders obvious this limitation.

- (d) **“a programmable clock controller having an embedded computer program therein, the computer program including instructions to:”**

Terrell teaches a programmable clock controller (clock controller 10) having an embedded computer program therein, the computer program including instructions. Ex. 1004, Claims 14, 19 (“a controller coupled to receive counts from the plurality of counter elements and which controls the common clock to have a frequency determined as a function of the counts from the plurality of counter elements”), (“wherein the ... controller [is] implemented under software control.”), Figure 3, ¶0020 (“software implementation ... for measuring and controlling a system clock”). Terrell describes that an “object of the present invention is to provide a software based [device] for ... controlling the system clock to operate at a minimum clock frequency which accommodates the operational requirements of the processors while reducing power consumed by them.” *Id.*, ¶¶0010-0011. Terrell describes “clock controller 10” as “a controller functional block” in its described “software implementation.” *Id.*, ¶0041.

Terrell teaches this limitation. Ex. 1002, ¶¶186-188.

As also discussed in Section XI.C.1(c), a POSA would have found it obvious to combine Terrell’s software based method and apparatus for controlling the system clock (programmable clock controller 10 with software implementation) with Chen’s clock gate logic (clock gate logic 24) for several reasons. Ex. 1002, ¶¶122-145.

First, as discussed, §XI.C.1(c), Terrell calls its control features “desirable” (“software based method and apparatus for ... controlling the system clock to operate at a minimum clock frequency,” Ex. 1004, ¶0011) for on-chip systems like Chen’s. *Id.*, ¶¶0006-0008. Ex. 1002, ¶138.

Second, using Terrell's software-based method and apparatus for clock control with Chen's clock gate logic would have involved nothing more than simple substitution of one known element (clock controller 10) with another (Chen's clock gate logic 24) to obtain predictable results (a software-based programmable clock controller that controls the clock frequency signal supplied to the master devices and the bus). A POSA would have been motivated to combine Terrell's clock signal change functionality with Chen's clock gate logic 24 because that element of Chen is already involved in implementing the clock signal change based on the request. *See* Ex. 1003, 3:20-22. Ex. 1002, ¶143.

Third, it would have been "obvious to try" using the software-based method and apparatus for controlling Terrell's clock (software based programmable clock controller) with Chen's clock gate logic circuit 24. Terrell recognizes a need for a control feature "to be able to reduce the frequency of a shared clock to the minimum frequency that allows the processing elements to function correctly while using the least amount of power." Ex. 1004, ¶0005.

Terrell acknowledges a finite number of solutions (two) for controlling the clock to accomplish this control feature: software-based and hardware-based. *Id.*, ¶0011. The '759 patent also recognizes the same two solutions for clock control: "[t]he functionality of various systems, modules, circuits, devices or components described herein may be implemented as hardware ... and/or software." Ex. 1001, 2:51-57. Ex. 1002, ¶¶143-144.

A POSA would have pursued Terrell's software-based clock control with a reasonable expectation of success because this is a simple modification of Terrell's software-based approach to Chen's clock gate logic and performs the same clock frequency signal control. This combination would have been obvious to try. Ex. 1002, ¶¶143-144.

Based on Terrell's disclosure, a POSA would have found it obvious to implement the functionalities set forth below, §§XI.C.2(e) to XI.C.3(b), as programmable instructions in software

on Chen's clock gate logic controller 24 as modified to be a programmable clock controller as taught by Terrell. Ex. 1002, ¶¶188, 192, 195.

Accordingly, Chen in view of Terrell renders these limitations obvious. Ex. 1002, ¶191.

(e) “receive the request provided by the first master device;”

As described, §§XI.C.1(b) and XI.C.2(d), Chen in view of Terrell renders obvious this limitation.

(f) “provide the clock frequency of the high-speed clock as an output to control a clock frequency of a second master device coupled to the bus in response to receiving the request provided by the first master device; and”

As described, §§XI.C.1(e) and XI.C.2(d), Chen in view of Terrell renders obvious this limitation.

(g) “provide the clock frequency of the high-speed clock as an output to control the variable clock frequency of the bus in response to receiving the request provided by the first master device.”

As described, §§XI.C.1(f) and XI.C.2(d), Chen in view of Terrell renders obvious this limitation, and accordingly renders claim 14 obvious. Ex. 1002, ¶191.

3. Claim 17

(a) “The system of claim 14,”

As described, §XI.C.2, Chen in view of Terrell renders claim 14 obvious.

(b) “wherein the instructions to provide the clock frequency of the high-speed clock as an output to control the variable clock frequency of the bus include instructions to adjust the clock frequency of the bus.”

Chen in view of Terrell renders obvious this limitation. As explained, §§XI.C.1(b) and XI.C.1(f), Chen controls the clock frequencies of bus 40 (and its devices 34, 36). Chen teaches, “Clock gate circuit 24 causes the frequency of bus 40 to be dynamically changed.” Ex. 1003, 3:20-21. Chen teaches adjusting the bus clock frequency between at least 50Mhz and 100Mhz. *Id.*, 3:44-45

(“operation of I/O bus 40 at 50 and 100Mhz”), 5:3-5 (“Clock gate logic 24 will ... drive bus 40 at 100Mhz....”). As explained, §XI.C.2(d), Chen’s clock gate logic controller 24 as modified into a programmable clock controller taught by Terrell would have performed this process to “adjust the clock frequency of the bus” by executing programmed instructions.

Chen in view of Terrell renders claim 17 obvious. Ex. 1002, ¶¶195-197.

D. Ground IV: Claims 18, 21-22, 24 are Obvious Over Chen in View of Terrell and Kiriake

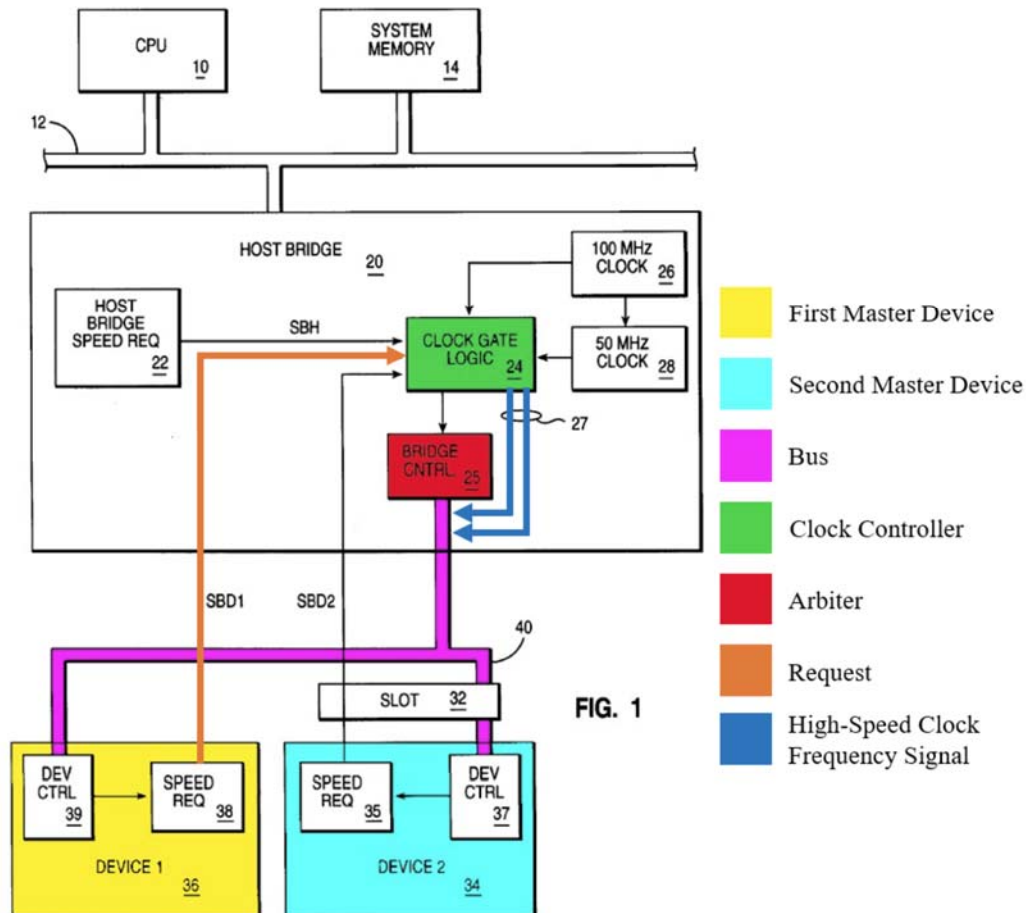
1. Claim 18

(a) “A system comprising: a bus capable of operation at a variable clock frequency”

As described, §XI.C.1(f), Chen teaches this. Chen teaches a system comprising a bus 40 capable of “operation ... at 50 and 100 Mhz” and that “other incremental increases in operating frequency are also contemplated ..., e.g. from 66 to 100 Mhz.” Ex. 1003, 3:44-48. Ex. 1046, ¶204.

(b) “a first master device coupled to the bus”

As described, §XI.C.1(a), Chen discloses a first master device (“Device 1” 36) coupled to a bus (I/O bus 40). Ex. 1003, Fig. 1, 2:67-3:3. Ex. 1046, ¶204.



- (c) “an arbiter coupled to the bus and coupled to the first master device, the arbiter configured to control flow of data on the bus; and”

As described, §XI.B.1(c), Kiriake teaches this. *See also* Ex. 1028, ¶¶0011, ¶¶0027; Fig. 1 (annotated below). Ex. 1046, ¶¶206-212.

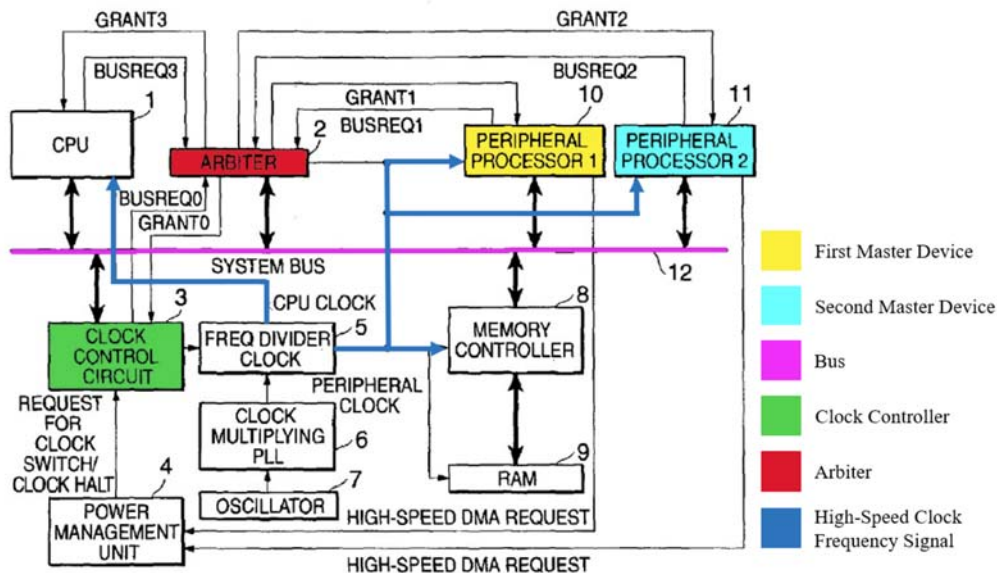


FIG. 1

It would have been obvious to modify Chen’s bridge control logic 25 in view of Terrell to include Kiriake’s arbiter. Chen’s bridge control logic “provides capability for the host bridge to perform transactions.” Ex. 1003, 3:17-20. Kiriake’s arbiter 2 is coupled to bus 12 and to master device 10 so the arbiter can recognize master device 10 as the bus master for a “bus transaction.” Ex. 1028, ¶¶0027-0028. Similarly, adding Kiriake’s arbiter 2 to Chen’s system in view of Terrell at bridge control logic 25 would have resulted in an arbiter (Kiriake’s arbiter 2) coupled to the bus (Chen’s I/O bus 40) and to the first master device (Chen’s device 36 in view of Terrell) to “provide[] capability for the host bridge to perform transactions.” Ex. 1003, 3:17-20, Fig. 1. A POSA would have been motivated to combine these with a reasonable expectation of success for several reasons. Ex. 1046, ¶¶189-200.

First, a POSA would have understood that Chen's teachings about arbitration cycles describe controlling initiation of bus traffic, and would have been motivated to incorporate other well-known teachings about arbiters (Kiriake's arbiter 2). Ex. 1046, ¶¶194-196. Chen's master devices already perform arbitration cycles for initiating transactions via the bus. Ex. 1003, 3:7-10. Chen teaches that bridge control logic 25 "provides capability for the host bridge to perform transactions." *Id.*, 3:17-20. Kiriake's arbiter similarly arbitrates initiation of bus transactions. Ex. 1028, ¶0027. It was well-known, in systems with multiple master devices on a bus, to add bus arbiter logic (of Kiriake) to a central bus interface or host bridge (of Chen) to control bus access. Ex. 1043, 5:64-65, 14:46-57, Fig. 2. Therefore, a POSA would have been motivated to combine Kiriake's arbiter with Chen's bridge control logic 25 to "perform[] arbitration relating to use of this system bus," Ex. 1028, ¶0011, and would have had a reasonable expectation of success in doing so. Ex. 1046, ¶¶194-200.

Second, Chen, Terrell, and Kiriake describe similar systems in the same field of endeavor and their teachings are complementary, as described §§X.A, X.B, and X.E. In the annotated figures below, all three references teach a plurality of master devices on a bus and a clock controller for adjusting the clock frequency of the systems for improved performance or power savings. *See e.g.*, Ex. 1003, 2:1-14, 5:61-64; Ex. 1004, ¶¶0002-0009; Ex. 1028, ¶¶0004, 0011, 0027. Thus, adding Kiriake's arbiter to Chen's bridge control logic 25 in view of Terrell would have involved nothing more than combining or substituting prior-art elements according to known methods (using Kiriake's arbiter to perform arbitration cycles) to yield predictable results (to arbitrate devices on Chen's bus). Ex. 1046, ¶¶189-200.

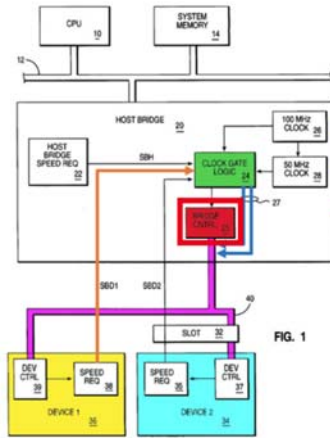


Figure 1 of Chen

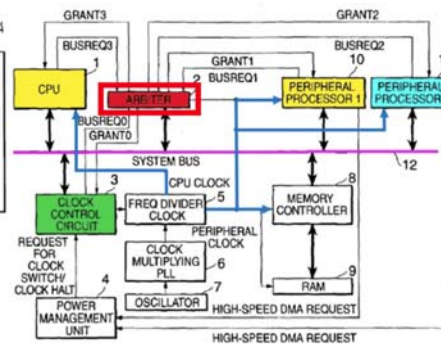


Figure 1 of Kiriake

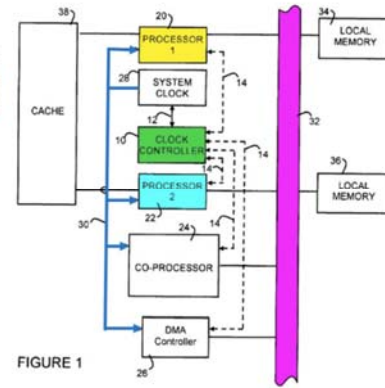


FIGURE 1

Figure 1 of Terrell

Finally, Terrell states that control features (such as those taught in Terrell and Kiriake) are “desirable” to include in on-chip bus systems like Chen’s system. Ex. 1004, ¶¶0005-0008. Kiriake also recognizes, “[T]o reduce power consumption, for example ... the clock for a circuit section in which high-speed operation is not required is switched to low speed.” Ex. 1028, ¶0004. As described, §XI.C.1(c), a POSA would have been motivated to modify Chen’s capability-based heuristic with Terrell’s need-based heuristic to select a clock frequency that increases the devices’ frequency only when needed, to reduce power consumption, even if the devices are capable of higher speeds. A POSA would have further been motivated to combine Kiriake’s control features (“arbiter that performs arbitration relating to use of this system bus”) with Chen’s bridge control logic 25 relating to use of I/O bus 40 in view of Terrell as the particular mechanism employed “to reduce power consumption,” see Ex. 1028, ¶¶0004, 0011, Ex. 1004, ¶¶0005-0008, and would have had a reasonable expectation of success in doing so. Ex. 1046, ¶¶129-152, 189-200.

(d) “a clock controller coupled to the arbiter and coupled to the first master device,”

Chen in view of Terrell and Kiriake renders obvious this limitation. Chen discloses a clock controller (clock gate logic 24) coupled to the first master device (“Device 1” 36). Ex. 1003, 3:13-17, Fig. 1. Just as Kiriake’s arbiter 2 is coupled to clock control circuit 3, *see* Ex. 1028, ¶¶0027, 0028,

0032, Fig. 1 (see red arrows between arbiter 2 and clock control circuit 3), adding Kiriake's arbiter to Chen's system in view of Terrell at bridge control logic 25 as described, §XI.D.1(c), would have resulted in the **arbiter (Kiriake's arbiter 2 combined with Chen's bridge control logic 25)** coupled to the **clock controller (Chen's clock gate logic 24)**. See Ex. 1003, 3:13-20, Fig. 1 (see red arrow between clock gate logic 24 and bridge control logic 25). Ex. 1046, ¶¶209, 214-218.

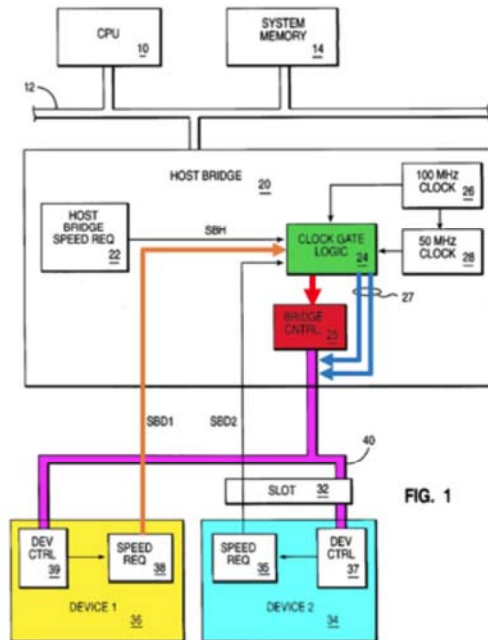


Figure 1 of Chen

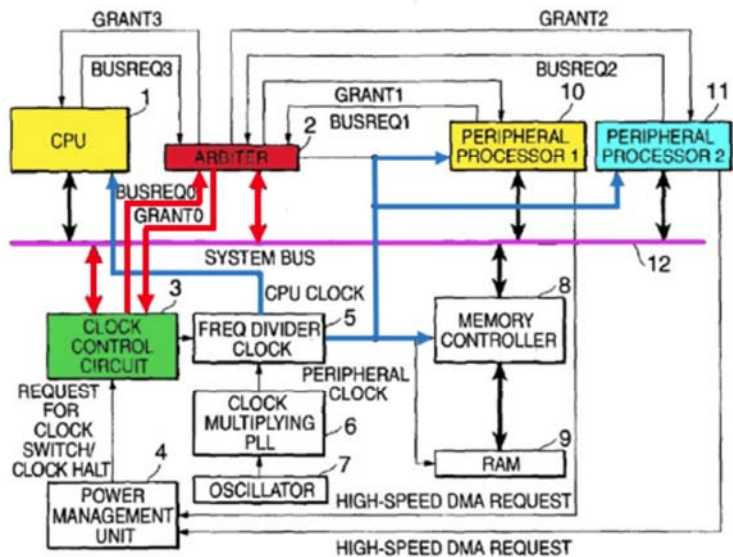


Figure 1 of Kiriake

As discussed, §XI.D.1(c), a POSA would have found it obvious to, and would have been motivated to, combine Kiriake's arbiter with Chen's system in view of Terrell, with a reasonable expectation of success. Ex. 1046, ¶¶189-200.

- (e) “the clock controller configured to output a clock frequency of a high-speed clock to control the variable clock frequency of the bus and to control a clock frequency of a second master device coupled to the bus,”

As described, §§XI.C.1(e) and XI.C.1(f), Chen discloses a clock controller (**clock gate logic 24**) configured to output a clock frequency of a high-speed clock (**clock line 27 output**) to control the variable clock frequency of the bus (**I/O bus 40**) and to control a clock frequency of a second

master device (device 34) coupled to the bus. Ex. 1046, ¶226.

- (f) **“the clock controller configured to receive a request to change the clock frequency of the high-speed clock from the first master device,”**

As described, §XI.C.1(b), Chen teaches a clock controller (clock gate logic 24) configured to receive a request to change a clock frequency of a high-speed clock (signal SBD of speed request circuit 38) from a first master device of the plurality of master devices (device 36). Ex. 1003, 3:61-64; Fig. 1. Ex. 1046, ¶226.

- (g) **“the request sent from the first master device in response to a predefined change in performance of the first master device”**

As described, §§XI.C.1(a) to XI.C.1(c), Chen in view of Terrell renders obvious this limitation. Ex. 1046, ¶226.

- (h) **“wherein the clock controller is configured to adjust the variable clock frequency of the bus in response to receiving the request from the first master device, and”**

As described, §§XI.C.1(e) and XI.C.1(f), Chen discloses this. Ex. 1003, 4:63-5:5 (“Clock gate logic 24 will then enable the high frequency clock 26 and drive bus 40 at 100Mhz.”), 5:59-65, Claim 13. Ex. 1046, ¶226.

- (i) **“wherein the predefined change in the performance is due to loading of the first master device as measured within a predefined time interval.”**

As discussed, §XI.C.1(d), Chen in view of Terrell renders obvious this limitation.

Chen in view of Terrell and Kiriake renders claim 18 obvious. Ex. 1046, ¶226.

2. Claim 21

- (a) **“The system of claim 18, wherein”**

As described, §XI.D.1, Chen in view of Terrell and Kiriake renders obvious claim 18.

(b) “adjusting the variable clock frequency of the bus comprises decreasing the clock frequency of the bus.”

Chen teaches “[c]lock gate circuit 24 causes the frequency of bus 40 to be dynamically changed (gated) by transmitting the appropriate device unique clock lines 27.” Ex. 1003, 3:20-22. Chen teaches “operation of I/O bus 40 at 50 and 100 Mhz.” *Id.*, 3:44-45. Chen therefore teaches decreasing the frequency of the bus from 100 Mhz to 50 Mhz. Ex. 1046, ¶236-238.

3. Claim 22

(a) “The system of claim 18, wherein”

As described, §XI.D.1, Chen in view of Terrell and Kiriake renders obvious claim 18.

(b) “adjusting the variable clock frequency of the bus comprises selecting the variable clock frequency to be a frequency divisible by a factor of 1, 2, 4, 8, or 16.”

Chen teaches adjusting the bus’s frequency to “50 and 100 Mhz.” Ex. 1003, 3:44-45. These frequencies are divisible by 16, etc. Ex. 1046, ¶¶243-246.

4. Claim 24

(a) “The system of claim 18, wherein”

As described, §XI.D.1, Chen in view of Terrell and Kiriake renders obvious claim 18.

(b) “the predefined change in the performance of the first master device comprises a variation in load of the first master device.”

Terrell teaches adjusting the system clock frequency based on a variation in the load of the master device. Ex. 1004, ¶0054 (“[D]uring a particular processing period the present invention may cause the system clock frequency to be lowered, at a later time any *increased processor activity can be recognized* by the present invention and the system clock frequency raised accordingly.”). Ex. 1046, ¶¶254-255.

XII. Conclusion

Based on the foregoing, claims 1, 14, 17-18, 21-22, 24 of the '759 patent are unpatentable, and Petitioner requests institution of an *inter partes* review to cancel them.

Respectfully submitted,

By: /s/ Andrew T. Oliver/

Andrew T. Oliver, Reg. No. 62,764

Appendix of Exhibits for U.S. Patent 7,725,759 – Petition for *Inter Partes* Review

Exhibit	Description
1001	U.S. Patent 7,725,759
1002	Declaration of Dr. Bruce Jacob, dated October 2019
1003	U.S. Patent 5,838,995 (“Chen”)
1004	U.S. Patent App. Pub. 2004/0098631 (“Terrell”)
1005	U.S. Patent 6,298,448 (“Shaffer”)
1006	U.S. Patent 7,360,103 (“Lint”)
1007	U.S. Patent App. Pub. 2005/0102560 (“Takatoshi”)
1008	U.S. Patent 4,670,837 (“Sheets”)
1009	THE ART OF COMPUTER SYSTEMS PERFORMANCE ANALYSIS, Jain, 1991
1010	June 29, 2005, U.S. Application Serial No. 11/170,475 (“475App”) as filed
1011	475App, Preliminary Amendment of July 27, 2006
1012	475App, Office Action 6/18/07
1013	475App, Amendment of September 17, 2007
1014	475App, Final Office Action of November 27, 2007
1015	475App, Amendment and Request for Continued Examination of February 27, 2008
1016	475App, Office Action 4/2/08
1017	475App, Response 4/29/08
1018	475App, Final Office Action 7/10/08
1019	475App, Amendment and Request for Continued Examination 9/10/08
1020	475App, Office Action 10/6/08
1021	475App, Response 12/26/08
1022	475App, Office Action 4/3/09
1023	475App, Response 6/25/09
1024	475App, Final Office Action 10/27/09
1025	475App, Response of 1/7/10
1026	475App, Notice of Allowance 1/28/10
1027	Jury Verdict
1028	U.S. Patent App. Pub. 2003/0159080 (“Kiriake”)
1029	E.P. Patent App. Pub. 0032818
1030	U.S. Patent 6,163,848
1031	U.S. Patent 6,643,792

Exhibit	Description
1032	U.S. Patent 6,813,719
1033	U.S. Patent 7,007,121
1034	U.S. Patent 7,155,631
1035	Grunwald, Policies for Dynamic Clock Scheduling, Proceedings of the 4th Symposium on Operating System Design and Implementation (OSDI '00), pp. 73–86, October 2000
1036	Pering, The simulation and evaluation of dynamic voltage scaling algorithms. In Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED'98), pp. 76–81, June 1998
1037	Varma, A control-theoretic approach to dynamic voltage scheduling, In Proc. International Conference on Compilers, Architecture, and Synthesis for Embedded Systems (CASES 2003), pp. 255–266, October 2003
1038	Weiser, Scheduling for reduced CPU energy, In Proc. First USENIX Symposium on Operating Systems Design and Implementation (OSDI 94), pp. 13–23, November 1994
1039	Ravishankar, Cache implementation for multiple microprocessors, Proceedings of IEEE COMPCON, pp. 346–350, February 1983
1040	Declaration of Dr. Sylvia Hall-Ellis
1041	U.S. Patent 4,789,926
1042	U.S. Patent 5,583,999
1043	U.S. Patent 5,987,614
1044	U.S. Patent 6,115,823
1045	Claim Construction Order
1046	Declaration of Dr. Bruce Jacob

CERTIFICATE OF COMPLIANCE REGARDING WORD COUNT

I hereby certify that the foregoing, Petition for *Inter Partes* Review of U.S. Patent 7,725,759, contains 13592 words as measured by the word processing software used to prepare the document, in compliance with 37 C.F.R. § 42.24 (d), and excluding the table of contents, mandatory notices, certificate of service, certificate of compliance regarding word count, and appendix of exhibits.

Respectfully submitted,

By: /s/ Andrew T. Oliver/

Andrew T. Oliver, Reg. No. 62,764

CERTIFICATE OF SERVICE

I hereby certify that on June 7, 2021, I caused a true and correct copy
of the foregoing materials:

- Petition for Inter Partes Review
- Certificate of Compliance Regarding Word Count
- OpenSky Industries LLC Power of Attorney
- Fee Authorization
- Exhibits 1001-1046

to be served via FedEx on the following attorney of record as listed on PAIR, by depositing same
with FedEx for overnight delivery:

Eric Scheuerlein or Janey Rodriguez Garcia
Daylight Law, P.C.
3790 El Camino Real
Unit #531
Palo Alto, CA 94306

Respectfully submitted,

By: /s/ Andrew T. Oliver/

Andrew T. Oliver, Reg. No. 62,764