Docket Nos. 2018-1439 (Lead), -1440, -1441, -1444, -1445

In the

United States Court of Appeals

For the

Federal Circuit

TECHNOLOGY PROPERTIES LIMITED LLC, PHOENIX DIGITAL SOLUTIONS LLC and PATRIOT SCIENTIFIC CORPORATION,

Plaintiffs-Appellants,

v.

HUAWEI TECHNOLOGIES CO., LTD., FUTUREWEI TECHNOLOGIES, INC., HUAWEI DEVICE CO., LTD., HUAWEI DEVICE USA INC., HUAWEI TECHNOLOGIES USA INC., ZTE CORPORATION, ZTE USA, INC., SAMSUNG ELECTRONIC CO., LTD., SAMSUNG ELECTRONICS AMERICA, INC., LG ELECTRONICS, INC., LG ELECTRONICS U.S.A., INC., NINTENDO CO., LTD. And NINTENDO OF AMERICA, INC.,

Defendants-Appellees.

Appeal from the United States District Court for the Northern District of California
Case Nos. 3:12-cv-03786-VC, 3:12-cv-03865-VC, 3:12-cv-03876-VC, 3:12-cv-03877-VC, 3:12-cv-03880-VC, and 3:12-cv-03881-VC · United States District Judge Judge Vince Chhabria.

NON-CONFIDENTIAL BRIEF OF APPELLANTS

DENISE M. DE MORY, ESQ.
ddemory@bdiplaw.com
HENRY C. BUNSOW, ESQ.
hbunsow@bdiplaw.com
AARON R. HAND, ESQ.
ahand@bdiplaw.com
LAUREN N. ROBINSON, ESQ.
lrobinson@bdiplaw.com
BUNSOW DE MORY LLP
701 El Camino Real
Redwood City, California 94063
(650) 351-7248 Telephone
(415) 426-4744 Facsimile

Attorneys for Plaintiffs-Appellants, Technology Properties Limited LLC, Phoenix Digital Solutions, LLC and Patriot Scientific Corporation

April 23, 2018

U.S. PATENT NO. 5,809,336
CLAIM 6

6. A microprocessor system comprising:

a central processing unit disposed upon an integrated circuit substrate, said central processing unit operating at a processing frequency and being constructed of a first plurality of electronic devices;

an entire oscillator disposed upon said integrated circuit substrate and connected to said central processing unit, said oscillator clocking said central processing unit at a clock rate and being constructed of a second plurality of electronic devices, thus varying the processing frequency of said first plurality of electronic devices and the clock rate of said second plurality of electronic devices in the same way as a function of parameter variation in one or more fabrication or operational parameters associated with said integrated circuit substrate, thereby enabling said processing frequency to track said clock rate in response to said parameter variation;

an on-chip input/output interface, connected between said central processing unit and an external memory bus, for facilitating exchanging coupling control signals, addresses and data with said central processing unit; and

an off-chip external clock, independent of said oscillator, connected to said input/output interface wherein said off-chip external clock is operative at a frequency independent of a clock frequency of said oscillator and wherein a clock signal from said off-chip external clock originates from a source other than said oscillator.

FORM 9. Certificate of Interest **Form 9**
Rev. 10/17

UNITED STATES COURT OF APPEALS FOR THE FEDERAL CIRCUIT

Technology Properties Limited, et al. **v.** Huawei Technologies Co., Ltd., et al.

Case No. 18-1439 (lead) with 18-1440, -1441, -1444, -1445 (consolidated)

CERTIFICATE OF INTEREST

Counsel for the:
☐ (petitioner) ■ (appellant) ☐ (respondent) ☐ (appellee) ☐ (amicus) ☐ (name of party)

Technology Properties Limited LLC, Patriot Scientific Corporation, Phoenix Digital Solutions LLC,

certifies the following (use "None" if applicable; use extra sheets if necessary):

1. Full Name of Party Represented by me	2. Name of Real Party in interest (Please only include any real party in interest NOT identified in Question 3) represented by me is:	3. Parent corporations and publicly held companies that own 10% or more of stock in the party
Technology Properties Limited LLC	Technology Properties Limited LLC	n/a
Patriot Scientific Corporation	Patriot Scientific Corporation	n/a
Phoenix Digital Solutions LLC	Phoenix Digital Solutions LLC	(1) Technology Properties Limited LLC; and (2) Patriot Scientific Corporation. Patriot Scientific Corporation is a publicly held company and owns 10 percent or more of the membership interest in Phoenix Digital Solutions LLC.

4. The names of all law firms and the partners or associates that appeared for the party or amicus now represented by me in the trial court or agency or are expected to appear in this court **(and who have not or will not enter an appearance in this case)** are:

See Attachment A.

FORM 9. Certificate of Interest **Form 9**
Rev. 10/17

5. The title and number of any case known to counsel to be pending in this or any other court or agency that will directly affect or be directly affected by this court's decision in the pending appeal. *See* Fed. Cir. R. 47. 4(a)(5) and 47.5(b). (The parties should attach continuation pages as necessary).

See Attachment B.

4/23/2018
Date

/s/ Denise De Mory
Signature of counsel

Please Note: All questions must be answered

Denise De Mory
Printed name of counsel

cc: All Counsel of Record Via Court's CM-ECF

Reset Fields

ATTACHMENT A

Current Counsel of Record for all Plaintiffs/Appellees:

TECHNOLOGY PROPERTIES LIMITED LLC;
PHOENIX DIGITAL SOLUTIONS LLC; and
PATRIOT SCIENTIFIC CORPORATION.

Henry Bunsow (SBN 60707)
hbunsow@bdiplaw.com
Denise De Mory (SBN 168076)
ddemory@bdiplaw.com
Aaron Hand (SBN 245755)
ahand@bdiplaw.com
Lauren Robinson (SBN 255028)
lrobinson@bdiplaw.com
BUNSOW DE MORY LLP
701 El Camino Real
Redwood City, CA 94063
Telephone: (650) 351-7248
Facsimile: (415) 426-4744

Previous Counsel of Record for Plaintiff/Appellee:

TECHNOLOGY PROPERTIES LIMITED LLC

Barry J Bumgardner
Edward Reese Nelson, III
John Paul Murphy
Stacy Greskowiak McNulty
(Terminated on 12/2/15)
Thomas Christopher Cecil
(Terminated on 12/2/15)
NELSON BUMGARDNER, PC
3131 W 7th St., Suite 300
Ft Worth, Tx 76107
817-377-9111
Email: barry@nelbum.com
Email: ed@nelbum.com
Email: murphy@nelbum.com
Email: stacie@nelbum.com
Email: tom@nelbum.com
TERMINATED: 08/11/2017

Michelle Gail Breit
Neustal Law Offices, LTD
2534 South University Drive, Suite 4
Fargo, ND 58103
(701) 281-8822
Fax: (701) 237-0544
Email: michelle@neustel.com
TERMINATED: 04/15/2015

William L. Bretschneider
Silicon Valley Law Group
50 W. San Fernando Street
Suite 750
San Jose, CA 95113
408-573-5700
Fax: 408-573-5701
Email: wlb@svlg.com
TERMINATED: 08/11/2017

Michael William Stebbins
Silicon Valley Law Group
One North Market Street
Suite 200
San Jose, CA 95113
408-573-5700
Fax: 408-573-5701
Email: mws@svlg.com
TERMINATED: 08/11/2017

David L. Lansky
Philip William Marsh
Thomas T. Carmack (Terminated 4/1/15)
Vinh Huy Pham
Arnold & Porter Kaye Scholer LLP
3000 El Camino Real
Five Palo Alto Square, Suite 500
Palo Alto, CA 94306-3807
650-319-4500
Fax: 650-319-4700
Email: David.Lansky@apks.com (Inactive)
Email: Philip.Marsh@apks.com
Email: Tom.Carmack@apks.com
Email: vinh.pham@apks.com
TERMINATED: 04/16/2015

James Carl Otteson
Arnold & Porter Kaye Scholer LLP
1801 Page Mill Road
Suite 110
Palo Alto, CA 94304
650-798-2970
Fax: 415-356-3099
Email: James.Otteson@apks.com
TERMINATED: 04/16/2015

Previous Counsel of Record for Plaintiff/Appellee:

PHOENIX DIGITAL SOLUTIONS LLC

Barry J Bumgardner
Edward Reese Nelson, III
John Paul Murphy
Stacy Greskowiak McNulty
(Terminated on 12/2/15)
Thomas Christopher Cecil
(Terminated on 12/2/15)
NELSON BUMGARDNER, PC
3131 W 7th St., Suite 300
Ft Worth, Tx 76107
817-377-9111
Email: barry@nelbum.com
Email: ed@nelbum.com
Email: murphy@nelbum.com
Email: stacie@nelbum.com
Email: tom@nelbum.com
TERMINATED: 08/11/2017

Christopher D. Banys
Jennifer Lu Gilbert
Richard Cheng-hong Lin
BANYS, P.C.
1030 Duane Avenue
Santa Clara, CA 95054
650-308-8505
Fax: 650-322-9103
Email: cdb@banyspc.com
Email: jlg@banyspc.com
Email: rcl@banyspc.com
TERMINATED: 06/05/2017

Christopher J Judge
BANYS PC
1032 Elwell Ct., No. 100
Palo Alto, CA 94303
650-308-8505
TERMINATED: 06/05/2017

David L. Lansky
Philip William Marsh
Thomas T. Carmack
(Terminated 4/1/15)
Vinh Huy Pham
ARNOLD & PORTER KAYE
SCHOLER LLP
3000 El Camino Real
Five Palo Alto Square, Suite 500
Palo Alto, CA 94306-3807
650-319-4500
Fax: 650-319-4700
Email: David.Lansky@apks.com (Inactive)
Email: Philip.Marsh@apks.com
Email: Tom.Carmack@apks.com
Email: vinh.pham@apks.com
TERMINATED: 04/16/2015

Eric Miller Albritton
ALBRITTON LAW FIRM
P.O. Box 2649
Longview, Tx 75606
903-757-8449
Fax: 903-758-7397
Email: ema@emafirm.com
TERMINATED: 08/11/2017

James Carl Otteson
ARNOLD & PORTER KAYE
SCHOLER LLP
1801 Page Mill Road, Suite 110
Palo Alto, CA 94304
650-798-2970
Fax: 415-356-3099
Email: James.Otteson@apks.com
TERMINATED: 04/16/2015

Michelle Gail Breit
Neustal Law Offices, LTD
2534 South University Drive, Suite 4
Fargo, ND 58103
(701) 281-8822
Fax: (701) 237-0544
Email: michelle@neustel.com
TERMINATED: 04/15/2015

Previous Counsel of Record for Plaintiff/Appellee:

PATRIOT SCIENTIFIC CORPORATION

Charles T. Hoge
Kirby Noonan Lance & Hoge LLP
350 Tenth Avenue, Suite 1300
San Diego, California 92101
Phone: (619) 231-8666
Fax: (619) 231-9593
choge@knlh.com
TERMINATED: 08/11/2017

ATTACHMENT B

Appealed from the Northern District Court:

Tech. Properties Limited v. Huawei, Case No. 3:12-cv-03865

Tech. Properties Limited v. ZTE Corp, Case No. 3:12-cv-3876

Tech. Properties Limited v. Samsung, Case No. 3:12-cv-3877

Tech. Properties Limited v. LG Electronics, Case No. 3:12-cv-3880

Tech. Properties Limited v. Nintendo, Case No. 3:12-cv-3881

Consolidated with Court of Appeals, Federal District:

Tech. Properties Limited v. Huawei, Case No. 18-1438 (LEAD)

Tech. Properties Limited v. ZTE Corp, Case No. 18-1440

Tech. Properties Limited v. Samsung, Case No. 18-1441

Tech. Properties Limited v. LG Electronics, Case 18-1444

Tech. Properties Limited v. Nintendo, Case No. 18-1445

TABLE OF CONTENTS

CONFIDENTIAL MATERIAL OMITTED

An excerpt from an architecture diagram for a Qualcomm 45nm HF PLL was omitted from page 28. The document from which this diagram was excerpted was designated as confidential by Defendants-Appellees in the underlying proceedings and was filed under seal pursuant to an order in the district court (Docket No. 146 in Case No. 3:12-cv-3865-VC).

TABLE OF AUTHORITIES

Page

CASES

STATUTES

RULES

STATEMENT OF RELATED CASES

The present appeal arises out of a summary judgment ruling which granted summary judgment of non-infringement of U.S. Patent No. 5,809,336 ("the '336 Patent") in these five Northern District of California cases:

- No. 3:12-cv-03865-VC, *Technology Properties Limited LLC, et al. v. Huawei Technologies Co., Ltd., et al*.;
- No. 3:12-cv-03876-VC, *Technology Properties Limited LLC, et al. v. ZTE Corporation, et al.*;
- No. 3:12-cv-03877-VC, *Technology Properties Limited LLC, et al. v. Samsung Electronics, Co., Ltd., et al.*;
- No. 3:12-cv-03880-VC, *Technology Properties Limited LLC, et al. v. LG Electronics, Inc., et al.*; and
- No. 3:12-cv-03881-VC, *Technology Properties Limited LLC, et al. v. Nintendo Co., Ltd., et al.*

Notices of Appeal were filed in these district court cases on January 5, 2018. The appeals were docketed on January 22, 2018, and consolidated when docketed; the *Huawei* case (No. 18-1439) was designated as the lead appeal:

- No. 18-1439, *Technology Properties Limited v. Huawei Technologies Co., Ltd*.;
- No. 18-1440, *Technology Properties Limited v. ZTE Corporation*;
- No. 18-1441, *Technology Properties Limited v. Samsung Electronic Co., Ltd*.;
- No. 18-1444, *Technology Properties Limited v. LG Electronics, Inc*.; and
- No. 18-1445, *Technology Properties Limited v. Nintendo Co., Ltd*.

These civil actions were previously the subject of appeals filed in this Court following the district court's issuance of a claim construction order and associated stipulated judgements of non-infringement in view of that order:

- No. 16-1306, *Technology Properties Limited LLC, et al. v. Huawei Technologies Co., Ltd., et al*.;
- No. 16-1307, *Technology Properties Limited LLC, et al. v. ZTE Corporation, et al*.;
- No. 16-1309, *Technology Properties Limited LLC, et al. v. Samsung Electronics, Co., Ltd., et al*.;
- No. 16-1310, *Technology Properties Limited LLC, et al. v. LG Electronics, Inc., et al*.; and
- No. 16-1311, *Technology Properties Limited LLC, et al. v. Nintendo Co., Ltd., et al*.

These prior appeals were consolidated on December 16, 2015, and the *Huawei* case (No. 16-1306) was designated as the lead appeal. Because the district court erred in a portion of its construction of "entire oscillator," this Court vacated and remanded in its decision dated March 3, 2017 in an opinion authored by Judge Moore (with Judges Wallach and Chen).

JURISDICTIONAL STATEMENT

This consolidated appeal arises from the civil actions for patent infringement of U.S. Patent No. 5,809,336 ("the '336 Patent") filed by Plaintiffs-Appellants against Defendants-Appellees in the U.S. District Court actions identified above. The district court had subject matter jurisdiction under 28 U.S.C. §§ 1331 and 1338(a).

The district court entered an order granting Defendants' Motion for Summary Judgment, finding that the accused products did not infringe Claims 6, 7, 9, 13, 14, and 15 (the "Asserted Claims") of the '336 Patent and entered Judgment in favor of Defendants on December 13, 2017. Appellants appeal the district court's Order and Judgment. Appellants represent that the Judgment is final.

Appellants' Notice of Appeal was timely filed on January 5, 2018 pursuant to Fed. R. App. 4(a)(1)(A).

This Court has jurisdiction over the appeals pursuant to 28 U.S.C. § 1295.

STATEMENT OF THE ISSUES

1. Whether the claimed ring oscillator disclosed in the '336 Patent specification was properly found to be disclaimed under Supreme Court authority?

2. Whether the district court erred by importing functional limitations into the asserted apparatus claims?

3. Whether the district court erred in holding that the asserted apparatus claims were not infringed where there were, at a minimum, disputed questions of fact regarding whether the accused ring oscillators meet the "entire oscillator" limitation?

STATEMENT OF THE CASE

I. INTRODUCTION

This is a case about the manner in which the judicially created doctrine of "prosecution disclaimer" has morphed to reach a result that was neither intended nor contemplated in the advent of disclaimer jurisprudence: a summary finding of noninfringement as to products that fall squarely within the language of the issued apparatus claims, based on statements in the prosecution history that relate to a proposed combination that did not result in an issued claim. Remarkably, the alleged disclaimers have been used to reach the conclusion that the accused "ring oscillator" does not meet the "entire oscillator" limitation of the asserted apparatus claims even though ***a "ring oscillator" is exactly the structure disclosed in the***

patent as the claimed "entire oscillator," and a "ring oscillator" is the very structure Applicants were arguing was allowable over the prior art. The doctrine has expanded too far beyond Supreme Court precedent. The district court's judgment should be reversed.

This case came before this Court last year after Appellants challenged the district court's claim construction order, which included a disclaimer finding. This Court: found that the district court erred in its disclaimer findings; applied its precedent to alter the scope of the disclaimer based on Applicants' discussions of the prior art (even though "[t]he patentee's disclaimer may not have been necessary" to distinguish the prior art); and remanded. On remand this Court's construction was applied in a manner that goes well beyond Supreme Court precedent.[1]

The doctrine of "disclaimer" traces its origins to at least as early as 1880 in *Goodyear Dental Vulcanite Co. v. Davis.*[2] At the time of *Goodyear Dental*, applicants could amend the patent specification during prosecution to provide the public with clear notice of the scope of their inventions. In *Goodyear Dental*, the Supreme Court held that a clear and unambiguous "disclaimer" occurred if a

[1] As explained *infra*, this Court can revisit this issue on appeal, and the issue is of such importance that it should be considered.

[2] *Goodyear Dental Vulcanite Co. v. Davis*, 102 U.S. 222 (1880).

patentee amended the patent specification to say: "I do not claim *x*." In view of such a clear and unequivocal statement (which in *Goodyear Dental* was accompanied by claim amendments commensurate in scope with the disclaimer), the Supreme Court determined in its infringement analysis that the patentee's disclaimer precluded a finding of infringement. In essence, the Supreme Court held that patentee could not reclaim, through an accusation of infringement, an embodiment that it expressly disclaimed via amendment to obtain the patent.

As initially applied and adopted by the Supreme Court, the limits of the disclaimer doctrine were clear and consistent with the public notice function of the patent grant: to provide public notice of the metes and bounds of the patentee's monopoly. Supreme Court authority is in accord with the clear rule that ***only*** an amendment to the patent itself (in either the specification or the claims) can qualify as a disclaimer.[3] Consistent with this authority, in the seminal *en banc* decision of *Markman v. Westview Instruments, Inc.*, this Court quoted *Goodyear Dental* when

[3] Later, the Patent Act changed such that it was no longer possible to amend the specification; instead, only claim amendments were allowed. Consistent with this change, the Supreme Court held that disclaimer applied to changes made to the claims during prosecution. *Schriber-Schroth Co. v. Cleveland Tr. Co.*, 311 U.S. 211, 217–18 (1940) ("Where the patentee in the course of his application in the patent office has, by amendment, cancelled or surrendered claims, those which are allowed are to be read in the light of those abandoned and an abandoned claim cannot be revived and restored . . .").

it reiterated that the prosecution history may provide insight into claim meaning, but cannot be used to "enlarge, diminish, or vary" their scope.[4]

Over time, the disclaimer doctrine has been greatly expanded by appellate and district courts, but not by the Supreme Court. Unlike the parallel doctrine of "prosecution history estoppel," which is applied during the doctrine of equivalents infringement analysis, an accused infringer is now able to invoke the disclaimer doctrine during claim construction—regardless of whether there is any alleged ambiguity in a claim or whether the accused infringer (or anyone) has ever relied on the prosecution statements. And now, amendments are no longer the hallmark of disclaimer, but instead an array of statements can qualify as a disclaimer. Statements made by an applicant during prosecution can now be found to be a disclaimer regardless of whether they have any impact on the examiner's decision to allow the patent claims, or whether they are made to overcome a prior art rejection. Now, as occurred here, disclaimer is applied as a means to graft additional limitations into the claims, including negative limitations. That the doctrine has expanded too far is evident in this case: an accused oscillator that is structurally identical to the oscillator disclosed in the specification has been

[4] *Markman v. Westview Instruments, Inc.*, 52 F.3d 967, 980 (Fed. Cir. 1995) (en banc), *aff'd*, 517 U.S. 370 (1996) (quoting *Goodyear Dental Vulcanite Co. v. Davis*, 102 U.S. 222, 227 (1880)).

summarily adjudicated to be non-infringing because the district court found that the oscillator did not function as the court believed was required when it imposed functional limitations not required by the claimed structural elements based on the negative limitations imposed by this Court.

Under Supreme Court authority, Appellants assert that the only clear disavowal of claim scope occurred by amendment when Applicants intentionally limited their claims: from a broad claim that required only a central processing unit ("CPU") and a ring oscillator to clock the CPU on the same integrated circuit; to a more limited claim that requires a CPU and a ring oscillator to clock the CPU on the same integrated circuit ***and*** (1) an input/output interface ("I/O interface") and (2) a second clock, external to the integrated circuit, to separately clock I/O interface. In fact, the "entire oscillator" limitation was never amended to reflect the disclaimers applied here, but ***was allowed as written*** when Applicants narrowed the claims to also require the I/O interface and the second clock. Appellants, represented by new counsel in this appeal, acknowledge that for the Court to reach this result, it must revisit its prior claim construction, but Appellants contend that important legal considerations and the potential for injustice warrant doing so.

Even if this Court does not revisit disclaimer, the district court's entry of judgment should still be reversed. It is a well-settled maxim that "apparatus claims

cover what a device is, not what a device does."[5] Notwithstanding this principle, the district court misapplied this Court's construction to import functional limitations into apparatus claims, and entered judgment ***based on its understanding of how the accused products work***.[6] The district court's approach was wrong as a matter of law. But at a minimum, there are disputed questions of fact that preclude entry of summary judgment. The district court improperly ignored conflicting expert testimony regarding whether the accused structures meet the limitations of the asserted apparatus claims.

Accordingly, Appellants respectfully request that this Court reverse and remand.

II. THE '336 INVENTIONS

It is August of 1989. The personal computer market is nascent. IBM compatible personal computers with 80386 processors (CPUs) operating at a speed of about 25 megahertz are the state of the art. There is a race amongst chip manufacturers to release faster and faster CPUs. Engineers Charles H. Moore and Russell Fish, the inventors of the '336 Patent, are developing next-generation microprocessors and are active participants in this race. Mr. Moore, already

---

[5] *Hewlett–Packard Co. v. Bausch & Lomb, Inc*., 909 F.2d 1464, 1468 (Fed. Cir. 1990) (holding *operational differences* inconsequential for apparatus; "apparatus claims cover what a device *is,* not what a device *does.*").

[6] Appx4-5.

known for inventing the Forth programming language, turned his attention to designing high performance CPUs. He and Mr. Fish ultimately develop the "Sh-boom" microprocessor, which was later inducted into the I.E.E.E. Chip Hall of Fame because of its advanced architecture. Appellants would go on to sell millions of dollars' worth of embodying products.[7]

As shown in Figure 1 of the '336 Patent, the invention at issue here was borne out of the "Sh-boom" microprocessor development efforts.[8] On August 3, 1989, Moore and Fish filed a comprehensive patent application describing many improvements to microprocessor architecture; seven patents ultimately issued from this application. Two distinct portions of the specification are significant here. The first describes only a single clock and a microprocessor and is entitled "Optimal CPU Clock Scheme," described in cols.

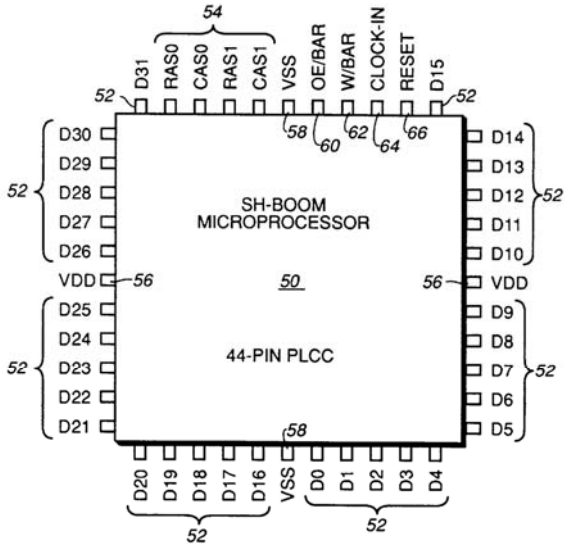


FIG._1

[7] Appx5391.

[8] Appx64.

16:43-17:10 of the '336 Patent.[9] The second portion describes a microprocessor and two clocks and is entitled "Asynchronous/Synchronous CPU," described in col. 17:11-37.[10]

The originally filed independent claims required only ***the combination of a microprocessor and a ring oscillator on the same chip***. As detailed below, the Examiner repeatedly rejected these claims based on two references, Sheets and Magar, that are single-clock references. ***Every statement*** that has been found to be a disclaimer in this case was made in Applicants' efforts to convince the Examiner to issue broad claims that required only the combination of a microprocessor and a single clock.

Ultimately, Applicants were unable to persuade the Examiner to issue the claims requiring only the Optimal CPU Clock Scheme, so they changed course, and amended the independent claims to also require a second (off-chip) clock and an I/O interface. The claims, including Claim 6 at issue here, were then allowed.

[9] Appx90-91.

[10] Appx91.

A. Optimal CPU Clock Scheme[11]

The specification describes an "Optimal CPU Clock Scheme" as one in which the clock is a "familiar ring oscillator" and "is fabricated on the same silicon chip as the rest of the microprocessor 50 [depicted above]."[12] The patent describes the advantages as follows: "The ring oscillator 430 is useful as a system clock … because its performance tracks the parameters which similarly affect all other transistors on the same silicon die."[13] Applicants described one advantage of manufacturing the CPU's clock and the CPU on the same chip: "For example, if the processing of a particular die is not good resulting in slow transistors, the latches and gates on the microprocessor 50 will operate slower than normal. Since the microprocessor 50 ring oscillator clock 430 is made from the same transistors on the same die as the latches and gates, it too will operate slower (oscillating at a lower frequency), providing compensation which allows the rest of the chip's logic to operate properly."[14]

---

[11] Relevant background information about the underlying technologies is found in Appellants' prior opening brief to this Court at Appx6703-6709, and the declaration of Appellants' expert, Dr. Oklobdzija at Appx6528-6541.

[12] Appx90 at 16:56-58. "CPU" and "microprocessor" are used interchangeably in the '336 Patent and in this brief.

[13] Appx90 at 16:63-67.

[14] Appx91 at 17:2-10.

B. Asynchronous/Synchronous CPU

In addition to the improvement of the Optimal CPU Clock Scheme, the inventors realized that existing microprocessor architecture suffered from another significant limitation: most microprocessors derived all system timing from a single clock. As a result, different parts of the system could slow all other operations.[15] For example, to communicate with an output device (such as a printer), the entire microprocessor speed would slow down to match the printer's communication speed on the I/O interface.[16] To ensure that the microprocessor remained in synchronization with the I/O interface and external devices, a single external crystal oscillator was historically employed to handle clocking.[17]

The inventors made a pathbreaking decision—they decoupled the system clock used for the microprocessor from the clock used for input-output ("I/O") operations.[18] Moore and Fish called this ground-breaking improvement "Asynchronous/Synchronous CPU."[19] This ***asynchronous dual-clock scheme*** allowed for the use of the "Optimal CPU Clock Scheme" with the use of a

[15] Appx91 at 17:12-14.

[16] Appx91 at 17:14-17.

[17] Appx91 at 17:32-34.

[18] *E.g.*, Appx91 at 17:12-14, 17:32-34; *see also* Appx84 at 3:26-35.

[19] Appx91 at 17:11.

conventional crystal clock for I/O features.[20] Decoupling the system clock from the clock used for I/O features allowed for optimal microprocessor speeds even during slower I/O operations.[21]

Claims 6, 7, 9, and 13-15 of the '336 Patent were asserted in this case; each is an apparatus claim.[22] Claim 6, directed to the asynchronous dual-clock architecture, is representative:

> 6. A microprocessor system comprising:
>
> a central processing unit disposed upon an integrated circuit substrate, said central processing unit operating at a processing frequency and being constructed of a first plurality of electronic devices;
>
> an entire oscillator disposed upon said integrated circuit substrate and connected to said central processing unit, said oscillator clocking said central processing unit at a clock rate and being constructed of a second plurality of electronic devices, thus varying the processing frequency of said first plurality of electronic devices and the clock rate of said second plurality of electronic devices in the same way as a function of parameter variation in one or more fabrication or operational parameters associated with said integrated circuit substrate, thereby enabling said processing frequency to track said clock rate in response to said parameter variation;
>
> an on-chip input/output interface, connected between said central processing unit and an *off-chip* external memory

[20] Appx91 at 17:14-21; *see also* Appx90 at 16:54-67.

[21] Appx91 at 17:32-33.

[22] Appx112-113.

> bus, for facilitating exchanging coupling control signals, addresses and data with said central processing unit; and
>
> an *off-chip* external clock, independent of said oscillator, connected to said input/output interface wherein said *off-chip* external clock is operative at a frequency independent of a clock frequency of said oscillator *and wherein a clock signal from said off-chip external clock originates from a source other than said oscillator*.[23]

III. THE FILE HISTORY

A. The Original Independent Claims Attempted to Claim Only the "Optimal CPU Clock Scheme"

All of the independent claims in the initial application required only a CPU and a single clock:

> 19. A microprocessor system, comprising a central processing unit and a ring counter variable speed system clock connected to said central processing unit, said central processing unit and said ring counter variable speed system clock being provided in a single integrated circuit.[24]

With reference to Figure 17 below, the original independent claims required only the combination (circled in blue) of a CPU (70) and a "ring oscillator variable speed clock" 430 in the same integrated circuit:[25]

---

[23] Appx112 (italics indicate language added in subsequent *ex parte* reexamination). Six examination requests targeted the '336 Patent: two were denied; four were instituted, resulting in two in reexamination certificates.

[24] *See* Appx2110-2119 at Appx2111 (reproduced without annotated amendments).

[25] Appx79.

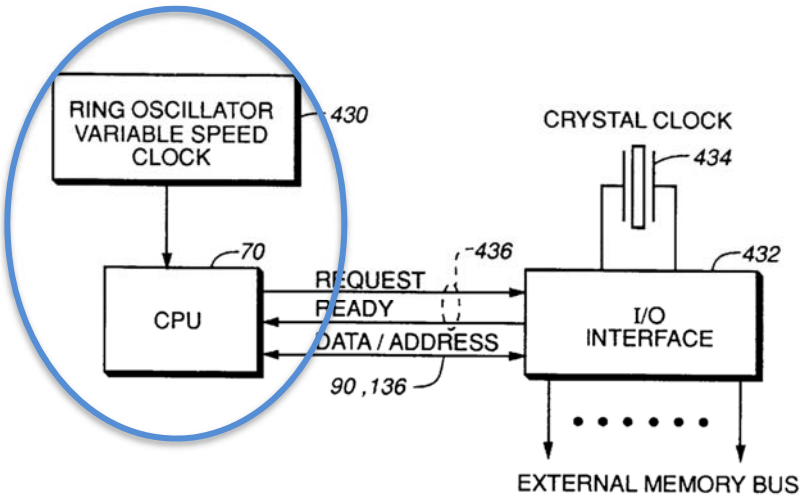


FIG._17

The application also included dependent claims that added the structures and connections in the rest of Figure 17, including, notably, a second clock:

> 20. The microprocessor system of claim 19 additionally comprising an input/output interface connected to exchange coupling control signals, address and data with said input/output interface, and ***a second clock independent of said ring counter variable speed system clock*** connected to said input/output interface.[26]

B. Applicants' First Arguments to Overcome Sheets

The Examiner rejected the independent claims on the basis of 35 U.S.C. Section 112.[27] In response, Applicants amended the independent claims to add that (1) the CPU and clock were in a single integrated circuit; (2) that the claimed clock

[26] Appx2111 (reproduced without annotated amendments) (emphasis added).

[27] *See* Appx2115.

was for clocking the CPU; and (3) that the CPU and the clock were made of electronic devices of like type.[28]

The Examiner also rejected the independent claims, 19 and 65, based on U.S. Patent No. 4,670,837 ("Sheets").[29] Sheets disclosed a microprocessor with a single variable-frequency clock, wherein the CPU was timed by an off-chip voltage-controlled oscillator ("VCO").[30] The CPU in Sheets monitored its own processing load and computed an appropriate operating frequency; the CPU then communicated the desired clock speed to the off-chip VCO by sending that information as a digital word across a data bus.[31] The Examiner also rejected dependent claim 20, which added the second clock, based on a combination of Sheets in view of Schaire.[32]

Applicants attempted to persuade the Examiner to issue the single-clock independent claims (including newly added claim 73, which matured into claim 6

[28] Appx2111-2112.

[29] Appx2116 ("The Examiner has rejected claims 19 and 65 under 35 U.S.C. § 103 as being unpatentable over Sheets.").

[30] Appx3496-3503.

[31] Appx3497 (fig.1), Appx3500 at 1:45-54, 2:57-68; Appx2127.

[32] Appx2124-2118 at Appx2118.

at issue in this case) over Sheets,[33] arguing that Sheets differed from the ring oscillator of then-pending claims because it relied on an *external clock*:

> The present invention does not similarly rely upon provision of frequency control information to an external clock, but instead contemplates providing a ring oscillator clock and the microprocessor within the same integrated circuit. The placement of these elements within the same integrated circuit obviates the need for provision of the type of frequency control information described by Sheets, since the microprocessor and clock will naturally tend to vary commensurately in speed as a function of various parameters (e.g., temperature) affecting circuit performance. Sheets' system for providing clock control signals to an external clock is thus seen to be unrelated to the integral microprocessor/clock system of the present invention.[34]

As to dependent claim 20, which already recited the second clock, Applicants responded that the combination of Sheets in view of Schaire did not teach the second clock: "Schaire fails to teach the claimed provision of separate, independent clock signals to an input/output interface buffer and microprocessor."[35]

[33] *See* Appx2110-2119 at Appx2112-2113, Appx2115-2117.

[34] Appx2117 (emphasis added).

[35] Appx2118.

C. Applicants Overcame the Sheets Rejection

The Examiner issued a Final Rejection on July 8, 1996. The Examiner and Applicants conducted telephone interviews in January 1997. Applicants summarized the conclusion of those calls:

> The above changes to the claims are based on the discussion in the interview. Proposed changes to claims 19, 65, and 73 were sent by facsimile to the Examiner on January 7 to facilitate the further discussion on January 8. On January 8, the Examiner agreed that these changes merited further consideration of the application and ***appeared to overcome the prior art of record***.[36]

The referenced "changes" to claim 73 (issued claim 6) were as follows:

> 73. (Amended). A microprocessor system comprising:
>
> a central processing unit disposed upon [a] an integrated circuit substrate, said central processing unit operating at a processing frequency and [including] constructed of a first plurality of [transistors] electronic devices;
>
> an oscillator disposed upon said integrated circuit substrate and connected to said central processing unit, said oscillator clocking said central processing unit at a clock rate and including a second plurality of [transistors] electronic devices, thus varying the [designed such that] operating characteristics of said first plurality and said second plurality of transistors [vary] in the same way as a function of parameter variation in one or more fabrication or operational parameters associated with said integrated circuit substrate, thereby enabling said processing

[36] Appx2124-2128 at Appx2126.

> frequency to track said clock rate in response to said parameter variation. [37]

Applicants explained: "Applicants are aware of no prior art teaching or suggesting a variable speed oscillator in the same integrated circuit with a microprocessor and clocking the microprocessor with a clock speed that varies correspondingly with changes in operating characteristics of electronic devices making up the microprocessor, as a result of being in the same integrated circuit as the microprocessor, as claimed."[38] In this same response, without conceding their single integrated circuit argument, Applicants made one of the statements this Court found to be a disclaimer (underlined below):

> Even if the Examiner is correct that the variable clock in Sheets is in the same integrated circuit as the microprocessor of system 100, that still does not give [sic] the claimed subject matter. In Sheets, a command input is required to change the clock speed. In the present invention, the clock speed varies correspondingly to variations in operating parameters of the electronic devices of the microprocessor because both the variable speed clock and the microprocessor are fabricated together in the same integrated circuit. <u>No command input is necessary to change the clock frequency</u>.[39]

---

[37] Appx2125. These amendments are not pertinent to the disclaimer at issue here.

[38] Appx2127 (emphasis added).

[39] Appx2127 (emphasis added; disclaimer underlined).

D. The Examiner Cites New Art: Magar

In April 1997, the Examiner again rejected the claims, but this time based on a new reference, U.S. Patent No. 4,503,500 ("Magar") as follows:[40]

> Shortly before issuing the Office Action, the Examiner had called to indicate that certain claims were allowable over the prior art, but when the undersigned attorney returned the Examiner's call, it was indicated that new prior art had been found and that a new office action would be forthcoming. It is assumed that the Magar reference relied on is the new prior art.[41]

Magar discloses a basic microprocessor that includes circuitry labeled "CLOCK GEN" on the same silicon substrate as the CPU. However, that CLOCK GEN circuity does not include any oscillator, and is itself incapable of frequency generation.[42] Instead, the CLOCK GEN circuitry in Magar has to be connected to an external oscillator, such as a crystal or other generator.[43] Applicants explained that "it is clear that the element in Fig. 17 missing from 2a in Magar is the ring counter variable speed clock 430, and that Magar is merely representative of 'most microprocessors' acknowledged as prior art in the above description from the present application, which prior art microprocessors use a 'conventional crystal

[40] Appx2042-2074.

[41] Appx2090-2097 at Appx2091.

[42] Appx2044 (fig.2a), Appx2067 at 15:23-41.

[43] Appx2067 at 15:26-28.

clock.'"[44] Applicants explained why their pending claims should be allowed over Magar:

> Because the variable speed clock is a primary point of departure from the prior art, independent claims 19, 65, 73 and 78 all recite a system including a variable speed clock or method including using a variable speed clock. In light of the prior art, of which Magar is a good example, Applicants are entitled to claims of this scope. Dependent claims 20, 66, 74, and 79 further recite a second clock, exemplified by the crystal clock 434 in Figure 17.[45]

Applicants also distinguished Magar's reliance on an external crystal for frequency generation:

> [O]ne of ordinary skill in the art should readily recognize that the speed of the cpu [sic] and the clock *do not* vary together due to manufacturing variation, operating voltage and temperature of the [integrated circuit] in the Magar microprocessor, as taught in the above quotation from the reference. ***This is simply because the Magar microprocessor clock is frequency controlled by a crystal which is also external to the microprocessor***.[46]

Indeed, Applicants explained that the crystal in Magar must be located off-chip: "Because of the cutting and trimming required, and that the crystal slice is typically suspended by two wires to allow it to freely oscillate, ***crystal oscillators***

[44] Appx2092.

[45] Appx2092.

[46] Appx2092-2093.

have never, to Applicants' knowledge, been fabricated on a single silicon substrate with a CPU, for instance."[47]

E. To Overcome Magar, Applicants Clarified That the "Entire Oscillator" Is on the Same Substrate as the CPU

The Examiner again rejected the claims based on Magar, this time in view of Pelgrom.[48] In response, Applicants and the Examiner again had a telephonic discussion. Applicants summarized this their explanation to the Examiner in this call as follows: "***Magar's clock generator relies on an external crystal connected to terminals X1 and X2 to oscillate***, as is conventional in microprocessor designs. It is not an entire oscillator in itself."[49] It was at this time that Applicants changed "an oscillator" in the independent claims to "an entire oscillator."[50]

Applicants, however, never obtained claims that required only the combination of a microprocessor and a single clock in the same chip.

F. The Claims Were Allowed When the Patentee Narrowed the Claims

The independent claims at issue issued when Applicants narrowed the claims to require all the elements on Figure 17. Regarding Claim 6, Applicants

---

[47] Appx2093 (emphasis added).

[48] Appx2099-2106 at Appx2101.

[49] Appx2101 (emphasis added).

[50] Appx2100.

added the limitations of then-pending dependent claim 74 into then-pending base claim 73 (73 and 74 are parallel to application claims 19 and 20 set forth above).[51] After the dependent limitations of claim 74 were incorporated into claim 73, the claim was allowed as Claim 6.

G. Other Portions of the File History Corroborate that Narrowing the Claims Resulted in Allowance[52]

After Applicants then pointed out in response to the Magar rejection that "[d]ependent claims 20, 66, 74 and 79 further recite a second clock, exemplified by the crystal clock 434 in Fig. 17,"[53] the Examiner responded as follows:

> Applicants states that Figure 17 of the instant application shows two clocks and that Magar's clock corresponds to applicant's I/O interface clock and therefore Mager [sic] does not show another clock 430 as shown in the drawing…. ***Note also that the claims do not call for two clock system***. Magar's clock clearly meets the claim limitation.[54]

[51] Appx2100; Appx2124-2128 at Appx2126.

[52] This section cites four file history excerpts that were not before the district court; Appellants request that this Court take Judicial Notice of them in its Motion for Judicial Notice ("MFJN"), filed herewith. That these amendments were the reason for allowance can be ascertained by comparing claim 73 in Paper 15 (Appx2099-2106) to the issued claims; the additional papers provide corroboration.

[53] Appx2092.

[54] MFJN004 (10/16/97 Office Action at 3 (Paper No 13)).

While the Examiner did not accept Applicants' arguments as to the patentability of the combination microprocessor and on-chip oscillator alone, during a telephonic interview, the Examiner "indicated that placing the limitations of dependent claims 20, 66, 75 and 79 into their respective parent claims would place the application in condition for allowance."[55] The Examiner subsequently allowed the revised claims—that cover the "Asynchronous/Synchronous CPU" that utilizes the "Optimal CPU Clock Scheme"—without further substantive amendment.[56]

IV. PRIOR CLAIM CONSTRUCTION AND STIPULATION

The '336 Patent has been the subject of multiple proceedings and ensuing claim construction opinions. Indeed, the "an entire oscillator disposed upon said integrated circuit substrate …" limitation found in Claim 6 (or variants thereof) has been interpreted (differently) many times.[57]

During claim construction proceedings in the underlying litigation, the district court initially construed the "entire oscillator" phrase to mean: "an [oscillator] located entirely on the same semiconductor substrate as the [central

[55] MFJN008 (4/29/98 Supplemental Amendment at 3 (Paper No. 16); MFJN009 (4/23/98 Examiner Interview Summary (Paper No 17)).

[56] MFJN010-012 (5/13/98 Notice of Allowability (Paper No. 18)) (the Examiner included "cleanup" amendments with the Notice of Allowability).

[57] Appx6718-6719.

processing unit] *that [1] does not require a control signal and [2] whose frequency is not fixed by any external crystal*."[58]

The newly imposed limitations narrowed the claims far beyond the scope of any statement made during prosecution. For example, the court rendered its pronouncement that that the entire oscillator "does not require a control signal" in the abstract, excluding the presence of *any* control signal whatsoever. Indeed, this construction was based on the finding that "unlike 'all cited references,' the claimed oscillator is ***completely free of inputs and extra components***."[59]

The blanket exclusion of *any* control signal (or any other input or component) from the oscillator arguably precluded any real-world application of the invention. To avoid the unnecessary expenditure of time or resources by the court or the parties, Appellants stipulated to non-infringement and entry of judgment to allow for immediate appellate review.[60]

V. THE PRIOR APPEAL

On a prior appeal, this Court held that "[t]he district court erred by holding that the patentee disclaimed any use of a command signal by the entire oscillator"

[58] Appx206-217 at Appx207 (emphasis added).

[59] Appx211(emphasis added).

[60] Appx4469-4471.

based on Sheets.[61] Rather, this Court found that Applicants argued that Sheets was different because its off-chip oscillator ***required*** "a command input ... to change the clock speed."[62] This Court went on to observe that in contrast, the patentees emphasized the benefits of their invention: "by placing the CPU and CPU clock on the same silicon substrate, the frequencies of both 'automatically vary together,'" but do "***not require*** manual or programmed inputs or external or extra components to do so."[63] Thus, this Court held that "an 'entire oscillator' is one 'that does not ***require*** a command input to change the clock frequency.'"[64]

When this Court affirmed the district court's imposition of the Magar-based disclaimer, it emphasized the reasoning behind that disclosure: "Throughout the prosecution history, the patentee argued Magar was distinguishable for two specific reasons: (1) it discloses a fixed-frequency crystal rather than a variable-frequency ring oscillator, and (2) it requires an external (off-chip) generator."[65] This Court concluded that Appellants surrendered two specific areas. "The first aspect of the patentee's disclaimer is that the 'entire oscillator' cannot be a fixed-

---

[61] *Tech. Properties Ltd. LLC v. Huawei Techs. Co*., 849 F.3d 1349, 1359 (Fed. Cir. 2017).

[62] *Id.* (quoting Appx2127).

[63] *Id.* at 1359-60 (emphasis added).

[64] *Id.* at 1360 (emphasis added).

[65] *Id.* at 1358.

frequency crystal oscillator."[66] As such, "the 'entire oscillator' must be a variable frequency oscillator rather than a fixed-frequency crystal."[67] Second, "the 'entire oscillator' cannot require an external crystal or frequency generator."[68] Whereas "Magar's clock generator relies on an external crystal … to ***oscillate***," "the '336 patent's entire oscillator was novel because 'it oscillates without external components (unlike the Magar reference).'"[69]

VI. THE ACCUSED PRODUCTS MEET THE STRUCTURAL LIMITATIONS OF THE ASSERTED APPARATUS CLAIMS

Each of the accused products includes a microprocessor that meets the structural limitations of asserted representative Claim 6—they have two clocks, a CPU, an I/O interface, and the required connections between these elements.[70]

The claimed "oscillator" is described in the specification as the "familiar ring oscillator … whose frequency is determined by temperature, voltage and process [("PVT")]."[71] As depicted in Figure 17, the claimed oscillator is also

---

[66] *Id.*

[67] *Id.*

[68] *Id.* at 1359.

[69] *Id.* (quoting Appx2102) (emphasis added).

[70] Appellants identified representative examples of the accused products' architecture and associated components. *See* Appx5250-5252.

[71] Appx90 at 16:56-60.

described as a "ring oscillator variable speed clock."[72] Claim 6 requires that it be fabricated on the same chip as the CPU, connected to the CPU and clock the CPU.[73]

It is undisputed that each accused product includes a variable speed ring oscillator (which is either a voltage-controlled oscillator ("VCO") or a current-controlled oscillator ("ISO") and interchangeably referred to herein as either the accused ring oscillator or VCO).

It is also undisputed that the accused VCOs operate at variable speeds. The accused VCOs are designed to output a range of frequencies.[74]

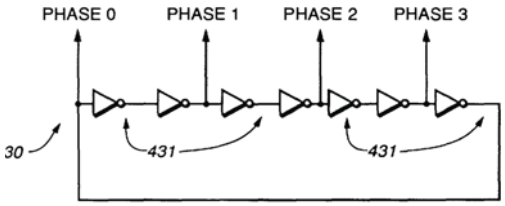
As depicted in Figure 18 of the patent (left) and the accused products (right), the claimed ring oscillator and the accused products have the same structure (an odd number of inversions in a loop):

[72] Appx79.

[73] Appx112 at 2:18-34 (Claim 6); *see also* Appx113 at 3:34-46 (Claim 13).

[74] Appx5299-300 ¶¶16, 18-19; *see also, e.g.*, Appx5592, Appx5600; Appx5694-5695; Appx5535-5538; Appx5551-5557; Appx5331 ¶50.

CONFIDENTIAL MATERIAL REDACTED



’336 Patent Figure 18

[redacted]

Accused Product Datasheet Excerpt[75]

It is undisputed that the accused VCOs are fabricated on the same substrate as the CPU.[76]

It is undisputed that the accused VCOs oscillate to provide the system clock signal that is supplied to the CPU.[77]

It is undisputed that the output frequency of accused VCOs vary based upon process, voltage, or temperature (“PVT”).[78]

Appellants identified these on-chip VCOs as the “entire oscillators” of the asserted claims.[79]

---

[75] *See, e.g.*, Appx5481-5482; Appx5538.

[76] *See* Appx5252; *see also id.* at n.6, Appx5026-5030 (Huawei), Appx5156-5164 (ZTE), Appx5177-5201 (Samsung), Appx5214-5225 (LG), Appx5238-5239 (Nintendo); Appx5287-5292 ¶4 (Qualcomm).

[77] *See, e.g.,* Appx5327 ¶43 (“The output of the VCO is used to drive the CPU.”); Appx5298-5300 ¶¶16-17.

[78] *See* Appx6539-6540; Appx5326, Appx5332-5333.

[79] *E.g.*, Appx5015-5016 (“The Ring Oscillator corresponds to the ‘entire oscillator’ in claim elements 6.c and 13.b because it is the part of the circuit that oscillates naturally in response to PVT.”); Appx5032; Appx5252.

VII. THE ACCUSED PRODUCTS ALSO INCLUDE A PLL THAT COMPENSATES FOR THE RING OSCILLATORS' VARIATIONS

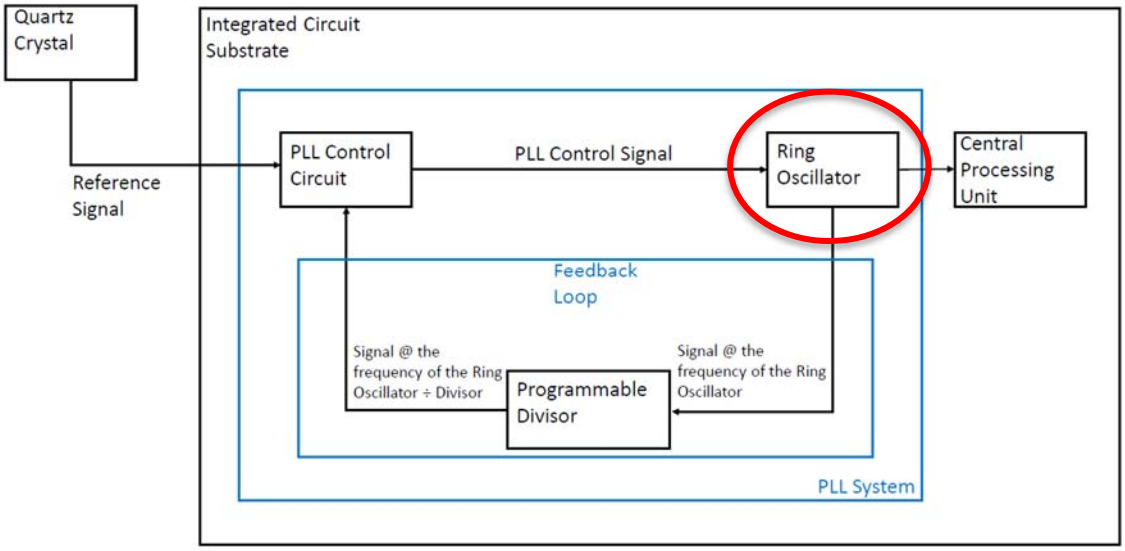
Because the accused ring oscillators in the accused products vary with fluctuations in PVT, the accused products include a structure known as a phase-locked loop or PLL to compensate for such variations. Indeed, Appellees' expert has admitted that the PLL compensates for variations in PVT: "Thus, the PLL Feedback Loop ***compensates for fabrication process, operational supply voltage and operational temperature ("PVT") variations*** to lock the VCO output frequency to a multiple of the fixed-frequency reference signal from the crystal oscillator."[80] Appellees' expert even analogized the PLL to the cruise control in a car, speeding up the car as needed based on conditions, such as going uphill (or in this case if the VCO slows based on conditions); or slowing the car down, for example as it goes downhill and begins to pick up speed.[81]

The parties agree that the basic structure of the accused devices, including the accused ring oscillator (circled in red) and the PLL, is as depicted below:[82]

[80] Appx5326 (emphasis added).

[81] Appx5332-5333.

[82] Appx5325-5327; Appx5251; Appx6508-6509.



VIII. SUMMARY JUDGMENT PROCEEDINGS

A. Appellees' Argument Based on the Sheets Disclaimer

Pursuant to the portion of this Court's claim construction relating to the Sheets disclaimer, the accused oscillator must be "***an oscillator*** located entirely on the same semiconductor substrate as the central processing unit *that does not require a command input to change the clock frequency*."[83]

Under the heading "[t]he accused ***oscillators*** require a command input to change frequency," the Appellees first argued that the accused products do not infringe because "the ***PLL*** is the antithesis of a variable speed clock. By its very nature and design, a ***PLL*** outputs a very stable fixed frequency."[84]

[83] *TPL*, 849 F.3d at 1360 (emphasis added).

[84] Appx5259 (citing Dr. Subramanian's ITC testimony, Appx5434 at 1213:5-10) (emphasis added).

Appellees next argued that "[e]mpirical evidence confirms that the ***PLL's*** frequency is fixed," relying on tests overseen by its expert while the accused products were operating. "The[] measurements confirm that the ***PLLs*** in the accused products do not vary as a function of PVT parameters, but rather output a stable fixed frequency."[85]

Finally, Appellees argued that "[a] 'command input' is required to change the output frequency value of the ***PLL*** in the accused products."[86] Appellees asserted: "The programmable divisors can be programmed to change the frequency of the ***PLL***. However, contrary to the Federal Circuit's construction, the ***PLL*** requires a command input to do so."[87]

Nowhere in Appellees' argument or supporting evidence did Appellees or their Declarants assert that the accused ***VCOs require*** a command input to change the clock frequency. Indeed, Appellees never even addressed the structure of the accused VCOs or their attributes; instead Appellees' expert simply asserted "the VCO in a ***PLL*** is not a free-running oscillator, ***rather it is an oscillator whose frequency is tightly controlled by the PLL***."[88]

---

[85] Appx5260 (emphasis added).

[86] Appx5264 (emphasis added).

[87] Appx5264-5265 (emphasis added).

[88] Appx5329 (emphasis added).

B. Appellees' Argument Based on the Magar Disclaimer

Pursuant to the portion of this Court's claim construction relating to the Magar disclaimer, the accused oscillator must be "an oscillator located entirely on the same semiconductor substrate as the central processing unit *whose frequency is not fixed by any external crystal*."[89]

Appellees argued "the undisputed operation of ***PLL***s generally and the empirical evidence of the operation of the PLLs in the accused products both confirm that the alleged 'entire oscillators' – *i.e*, the VCOs within the ***PLL***s in the accused products – output a stable fixed frequency."[90]

As Appellees' expert explained, the PLL in the accused products relies on an external crystal as a "reference signal," not to generate or fix the frequency:

> A "phase detector" inside the PLL Control Circuit block compares the phase of the feedback signal from the Feedback Loop and phase of the reference signal from the crystal oscillator. Specifically, the phase detector checks whether the feedback signal and the reference signal are in-phase with each other. If the two signals are not in-phase and the feedback signal's phase lags behind the reference signal's, the detector produces an output that instructs the VCO inside the PLL to go 'faster' in order to catch up. If the feedback signal's phase is ahead of the reference signal's phase, the detector's output will instruct the VCO to "slow down."[91]

---

[89] *TPL*, 849 F.3d at 1360 (emphasis added).

[90] Appx5274 (emphasis added).

[91] Appx5326.

Appellees argued that the accused products do not infringe because "PLL control circuitry compensates for variations in PVT parameters to ensure that the VCO output frequency is 'locked' to a fixed multiple of the reference signal from the crystal oscillator."[92] Notably, Appellees concede that the output of the PLL is not fixed but instead changes between a number of different frequencies.[93]

C. Appellants' Responses To Appellees' Summary Judgment Motion

Appellants first pointed out that the Appellees' arguments were flawed because the accused "entire oscillator" is the ring oscillator in the accused products, not the PLL.[94]

Appellants also argued that the accused VCOs met the "entire oscillator" limitation. Appellants' expert explained: "The addition of PLL circuitry to an on-chip ring oscillator does not change the ring oscillator's fundamental characteristics, which are the features determined by physics and nature—that its frequency varies based on fabrication or operational parameters, including process, temperature and voltage."[95] While Appellees' expert, Dr. Subramanian, never even made reference to this Court's construction,[96] Appellants' expert opined:

---

[92] Appx5274.

[93] Appx5253; Appx5293-5299 ¶16.

[94] Appx6497, Appx6508-6509.

[95] Appx6541 ¶38.

[96] *See generally* Appx5306-5309; Appx129, Appx132-133.

> It is my opinion that the ring oscillators in the accused products meet the Federal Circuit's definition of "entire oscillator." The frequencies generated by an on-chip ring oscillator in a PLL System necessarily depend on fabrication and operation parameters, such as process, voltage and temperature. Indeed, the PLL is employed in the accused products to manage the VCO because of its inherent variability to PVT — that is its purpose. Notwithstanding the presence of added PLL circuitry, the ring oscillator (VCO) will oscillate as soon as power is applied, and its frequencies vary based on fabrication and operational parameters….[97]

Appellants' expert explained that the crystal in the PLL is not used to generate a frequency (clock) for the CPU, but is instead only used as reference signal: "To be clear, the fixed external crystal reference, $F_{reference}$ does not produce the system clock, nor can this signal pass through the PLL circuitry. It is merely a sample to provide a basis for comparison. Likewise, the on-chip components do not rely on the off-chip crystal to generate a clock signal."[98] Appellees' expert confirmed that the crystal is used only as a reference signal.[99] Moreover, Appellees' expert and declarant concede that the accused VCOs oscillate to supply the clock for the CPU.[100]

---

[97] Appx6539 ¶33.

[98] Appx6539-6540 ¶34.

[99] Appx5326 ¶39.

[100] *See, e.g.*, Appx5327 ¶43 ("The output of the VCO is used to drive the CPU."); Appx5298-5300 ¶¶16-17.

IX. THE DISTRICT COURT'S SUMMARY JUDGMENT ORDER

The district court's analysis and reasoning are set forth in a five-page opinion, that begins:

> The plaintiffs ("TPL") stipulated to non-infringement under this Court's prior construction of the phrase "an entire oscillator disposed upon said integrated circuit substrate" as used in the asserted claims of Patent No. 5,809,336. The Federal Circuit then made a "minor modification" to that claim construction, holding that the proper construction of the disputed claim term is: "an oscillator located entirely on the same semiconductor substrate as the central processing unit that does not require a command input to change the clock frequency and whose frequency is not fixed by any external crystal." In doing so, the Federal Circuit noted that its change to the prior construction "likely does not affect the outcome in this case." The Federal Circuit's prediction was correct.[101]

In assessing whether the entire oscillators in the accused products required a "command input" to change their frequencies (i.e., the Sheets-based disclaimer), the district court focused on the PLL, not the accused entire oscillator.[102] Relying on Appellees' tests showing how the accused devices operate, the district court

[101] Appx4 (citations omitted).

[102] The district court's order suggested that Appellants "disavow[ed] at oral argument" a non-infringement position that the accused oscillators should be considered "in isolation." *See* Appx7. But, to the contrary, Appellants repeatedly clarified that the focus must be on the accused VCO, and that notwithstanding the presence of PLL circuitry, the accused VCO meets the claim limitation. *See, e.g.*, Appx136 ll.2-8.

analyzed how the PLL impacts the operation of the accused VCO and ultimately concluded that the '336 Patent required "free running oscillators," and that stabilizing effects of the added PLL circuitry therefore rendered the accused products non-infringing.[103]

Although the parties briefed the "fixed by an external crystal" limitation, the district court chose not to analyze or address those arguments—but nonetheless opted to speculate that "TPL would lose on that question as well."[104]

SUMMARY OF THE ARGUMENT

It is well-settled in patent law that the claims specifically point out and define the scope of the patentee's invention. The origin of the disclaimer doctrine involved amendments to the specification or claims that would place the public on notice of the scope of patentee's invention. These amendments were specific, explicit statements that made clear to the public what a patentee was not claiming: "I do not claim *x*" or a change in actual claims via amendment.

Consistent with those Supreme Court precedents, this Court's *Markman* decision reiterated that the prosecution history, while it may provide insight into the meaning of a claim, should not be used to alter the scope of issued claims. But despite that guidance, disclaimer doctrine has expanded to a degree that results in

[103] Appx8.

[104] Appx8.

inconsistent and unpredictable decisions about claim scope, conflicting with the public's ability to rely on issued claims and thereby undercutting the vital public notice function of issued patent claims.

This case exemplifies how far the disclaimer doctrine has strayed from its origins. Here, the doctrine of "prosecution disclaimer" has resulted in summary finding of noninfringement based on products that fall squarely within the language of the issued apparatus claims. The "disclaiming" statements were made to distinguish over the prior art on the grounds that the prior art did not disclose the claimed oscillator. Relying on those statements, this Court imposed negative limitations on the asserted claims. As a result of that construction, the accused oscillators—which are structurally identical to the oscillator described in the specification and that Applicant argued to the Examiner was not found in the prior art—have been deemed non-infringing.

Moreover, the accused oscillators have been summarily adjudicated to be non-infringing (of the asserted ***apparatus*** claims) because the district court found that the accused oscillators did not ***function*** as the district court believed was required under this Court's claim construction. The district court's approach was wrong as a matter of law. However, even absent the district court's improper focus on operation in its infringement analysis of apparatus claims, there are disputed questions of fact that preclude entry of summary judgment. The district court

improperly ignored conflicting expert testimony regarding whether the accused structures meet the limitations of the asserted apparatus claims. Even ***Appellees' own expert and witness declarations preclude summary judgment*** of noninfringement because they demonstrate that the accused devices meet the claimed structural limitations. Accordingly, Appellants respectfully request that this Court reverse and remand.

ARGUMENT

I. STANDARD OF REVIEW

A district court's grant of summary judgment is reviewed *de novo*.[105] Summary judgment is proper when all reasonable inferences have been drawn in favor of the non-movant and no genuine issue of material fact exists.[106] A district court reviewing summary judgment "is not 'to weigh the evidence and determine the truth of the matter but to determine whether there is a genuine issue for trial.'"[107] A district court's decision must draw all reasonable inferences in favor of the non-movant, even when "such inferences may create disputes regarding

---

[105] *Skedco, Inc. v. Strategic Operations, Inc.*, 685 F. App'x 956, 958 (Fed. Cir. 2017); *Mavrix Photographs, LLC v. Livejournal, Inc.*, 873 F.3d 1045, 1051 (9th Cir. 2017).

[106] Fed. R. Civ. P. 56(c); *Anderson v. Liberty Lobby, Inc.,* 477 U.S. 242, 248, 255 (1986).

[107] *Tolan v. Cotton*, 134 S. Ct. 1861, 1866 (2014) (citing *Anderson*, 477 U.S. at 249).

basic facts or regarding facts to be inferred from such facts."[108] When a reasonable jury examining the evidence of record can resolve a disputed fact in favor of either party, summary judgment cannot lie.[109]

Infringement determinations require two steps: (1) construing the claims for scope and meaning, and (2) comparing properly construed claims to the accused product.[110] A claim construction promulgated by this Court is considered the law of the case.[111] "[A] district court is without choice in obeying the mandate of the appellate court."[112] An appellate court, however, is empowered to revisit its own decisions; as Justice Holmes observed, the law of the case doctrine "merely expresses the practice of courts generally to refuse to reopen what has been decided, not a limit to their power."[113] This Court has recognized its authority to depart from the law of the case, including when a prior decision is "clearly

---

[108] *Absolute Software, Inc. v. Stealth Signal, Inc.*, 659 F.3d 1121, 1133 (Fed. Cir. 2011).

[109] *Id.*

[110] *Id.*

[111] *Ethicon Endo–Surgery, Inc. v. U.S. Surgical Corp.*, 149 F.3d 1309, 1315 (Fed. Cir. 1998).

[112] *In re Roberts*, 846 F.2d 1360, 1363 (Fed. Cir. 1988).

[113] *Messenger v. Anderson*, 225 U.S. 436, 444 (1912) (Holmes, J.); *see also S. Ry. Co. v. Clift*, 260 U.S. 316, 319 (1922).

incorrect and its preservation would work a manifest injustice."[114]

The second step of the infringement analysis is a question of fact.[115] A finding of infringement is warranted when each claimed element is found in an accused device.[116] "If a claim reads merely on a part of an accused device, that is enough for infringement."[117] "[I]n determining whether a product claim is infringed, … an accused device may be found to infringe if it is reasonably capable of satisfying the claim limitations, even though it may also be capable of non-infringing modes of operation."[118]

"On appeal from a grant of summary judgment of non-infringement, [this Court] determine[s] whether, after resolving reasonable factual inferences in favor of the patentee, the district court correctly concluded that no reasonable jury could find infringement."[119]

---

[114] *Intergraph Corp. v. Intel Corp*., 253 F.3d 695, 698 (Fed. Cir. 2001); *see also Gindes v. United States*, 740 F.2d 947, 950 (Fed. Cir. 1984) (applying a "stringent standard" requiring more than a mere suspicion of error).

[115] *Ethicon Endo-Surgery*, 149 F.3d at 1315.

[116] *See, e.g.*, *Cross Med. Prod. Inc. v. Medtronic Sofamor Danek, Inc.*, 424 F.3d 1293, 1310 (Fed. Cir. 2005) (citations omitted).

[117] *SunTiger, Inc. v. Sci. Research Funding Grp*., 189 F.3d 1327, 1336 (Fed. Cir. 1999) (citing *A.B. Dick*).

[118] *Hilgraeve Corp. v. Symantec Corp*., 265 F.3d 1336, 1343 (Fed. Cir. 2001) (citing *Intel Corp. v. United States Int'l Trade Comm'n*, 946 F.2d 821, 832 (Fed. Cir. 1991).

[119] *Absolute Software*, 659 F.3d at 1130.

II. JUDGMENT OF NONINFRINGEMENT BASED ON "PROSECUTION DISCLAIMERS" SHOULD BE REVERSED

A. The Claims and Specification Define the Limited Monopoly Granted to Inventors

"The patent law confers on the patentee a limited monopoly, the right or power to exclude all others from manufacturing, using or selling his invention. The claims of the patent control the metes and bounds of the limited monopoly. As the Supreme Court explained in 1877,

> Since the act of 1836, the patent laws require that an applicant for a patent shall not only, by a specification in writing, fully explain his invention, but that he '***shall particularly specify and point out the part, improvement, or combination which he claims as his own invention or discovery***.' This provision was inserted in the law for the purpose of relieving the courts from the duty of ascertaining the exact invention of the patentee by inference and conjecture, derived from a laborious examination of previous inventions, and a comparison thereof with that claimed by him. This duty is now cast upon the Patent Office. There his claim is, or is supposed to be, examined, scrutinized, limited, and made to conform to what he is entitled to.[120]

Consistent with this long-standing authority, "[i]t is a 'bedrock principle' of patent law that 'the claims of a patent define the invention to which the patentee is entitled the right to exclude.'[121]

---

[120] *Keystone Bridge Co. v. Phoenix Iron Co.*, 95 U.S. 274, 278 (1877).

[121] *Phillips v. AWH Corp.*, 415 F.3d 1303, 1312 (Fed. Cir. 2005) (citing *Innova/Pure Water, Inc. v. Safari Water Filtration Sys., Inc.*, 381 F.3d 1111, 1115

B. The Origins of Disclaimer Law

The doctrine that is now understood as a "prosecution disclaimer" traces its origins to at least as early as a case decided well over 100 years ago: *Goodyear Dental*.[122] *Goodyear Dental* dealt with a specific type of amendment to the specification, one in which the applicant literally wrote into the specification "I do not claim *x*":

> ***I do not claim*** the use of gutta-percha, or of any material which is merely rendered plastic by heat and hardened by cooling, in the manufacture of sets of artificial teeth; but what I do claim as my invention, and desire to have secured to me by letters-patent, is the improvement in the manufacture of sets of mineral or other artificial teeth, which consists in combining them with a rubber plate and gums, which (after the insertion of the teeth) are vulcanized by Goodyear's process, or any other process.[123]

Concurrent with these statements, the applicant amended his claims (additions underlined, deletions struck-through) commensurate in scope with the disclaimer

---

(Fed. Cir. 2004); *Vitronics Corp. v. Conceptronic, Inc.*, 90 F.3d 1576, 1582 (Fed. Cir. 1996) ("[W]e look to the words of the claims themselves … to define the scope of the patented invention"); *Markman v. Westview Instruments*, 52 F.3d 967, 980 (Fed. Cir. 1995) (en banc), *aff'd*, 517 U.S. 370 (1996) ("The written description part of the specification itself does not delimit the right to exclude. That is the function and purpose of claims.")).

[122] *See generally*, *Goodyear Dental Vulcanite Co. v. Davis*, 102 U.S. 222, 227 (1880) (accused dental plates did not infringe because applicant gave up certain subject matter based on "I do not claim" statements in the specification).

[123] *Id.* at 227 (emphasis added).

statement: "forming the plate and gums in which the teeth are inserted in one piece of hard rubber, or ~~other~~ vulcanite, i.e. an elastic material."

In its infringement analysis, the Supreme Court looked to the claims, and then to the applicant's statements to determine whether an accused product infringed. Though the patent is controlling, the Supreme Court indicated that it was permissible to consider whether the prosecution history confirmed an apparent disavowal, but cautioned against use of the prosecution history to alter a patent:

> We *do not mean* to be understood as asserting that *any correspondence* between the applicant for a patent and the Commissioner of Patents *can be allowed to enlarge, diminish, or vary the language of a patent* afterwards issued.[124]

Recognizing that the issued patent is intended to reflect the final memorial of the applicants' exchange with the Patent Office, Judge Learned Hand noted that under settled precedent, courts were not "to go through all that was said in the endless communications between applicant and Examiners to gather piecemeal the intent of the grant . . . ."[125]

Later, the Patent Act was amended and the practice of amending the specification was no longer allowed. Thereafter, the Supreme Court focused on

---

[124] *Goodyear Dental*, 102 U.S. at 227 (emphasis added).

[125] *Campbell Metal Window Corp. v. S.H. Pomeroy & Co.*, 300 F. 872, 873-74 (S.D.N.Y. 1924) (Hand, J.) (citing *Goodyear Dental*, 102 U.S. 222).

amendment of claims during prosecution: "Where the patentee in the course of his application in the patent office has, by amendment, cancelled or surrendered claims, those which are allowed are to be read in the light of those abandoned and an abandoned claim cannot be revived and restored to the patent by reading it by construction into the claims which are allowed."[126] The general rule was that "[c]laims may not be construed one way in order to obtain their allowance and in a different way against accused infringers."[127]

In 1995, this Court established the modern claim construction process in *Markman v. Westview Instruments*.[128] The claim construction inquiry begins with the claim language itself, followed by the specification in light of which the claims are read; if necessary, the prosecution history could be considered as a final form of intrinsic evidence.[129] And while the *Markman* Court permitted consultation of

---

[126] *Schriber-Schroth Co. v. Cleveland Tr. Co*., 311 U.S. 211, 217-18 (1940); *see also Graham v. John Deere Co*. *of Kansas City*, 383 U.S. 1, 33 (1966) ("It is, of course, well settled that an invention is construed not only in the light of the claims, but also with reference to the file wrapper or prosecution history. Claims as allowed must be read and interpreted with reference to rejected ones and to the state of the prior art …." (internal citation omitted)).

[127] *See, e.g.*, *Southwall Techs., Inc. v. Cardinal IG Co.*,, 54 F.3d 1570, 1576-77 (Fed. Cir. 1995) (citing *Unique Concepts, Inc. v. Brown*, 939 F.2d 1558, 1562 (Fed. Cir. 1991)).

[128] 52 F.3d 967 (Fed. Cir. 1995) (en banc), *aff'd*, 517 U.S. 370 (1996).

[129] *Markman*, 52 F.3d at 979.

the file history, it repeated the Supreme Court's century-old admonishment: "Although the prosecution history can and should be used to understand the language used in the claims, it too cannot 'enlarge, diminish, or vary' the limitations in the claims."[130]

C. Disclaimer No Longer Requires Amendment

By the early 2000's, the disclaimer doctrine had expanded such that it was no longer limited to claim amendments, but instead included "repeated statements" during prosecution when they (a) place the "public on notice of the invention's crucial feature" and (b) distinguish the claimed invention from the prior art to the extent that "it is not suitable to multiple interpretations."[131] A decade later, the doctrine had expanded such that disclaimer "analysis focuses on what the applicant said, not on whether the representation was necessary or persuasive."[132] It is now that case that an alleged "disclaimer" need not even be a statement that resulted in the patent grant, and "patentees may surrender more than necessary."[133]

---

[130] *Id.* at 980 (quoting *Goodyear Dental*, 102 U.S. at 227).

[131] *Omega Eng'g, Inc, v. Raytek Corp.*, 334 F.3d 1314, 1326-27 (Fed. Cir. 2003).

[132] *Uship Intellectual Properties, LLC v. United States*, 714 F.3d 1311, 1315 (Fed. Cir. 2013).

[133] *Tech. Properties Ltd. LLC v. Huawei Techs. Co.*, 849 F.3d 1349, 1359 (Fed. Cir. 2017).

D. Current Disclaimer Law Produces Inconsistent and Unpredictable Results

The consequences of departure from this clear rule are real. In the last two decades, the prosecution disclaimer opinions from varying panels of this Court have expressed divergent views and applications of the disclaimer doctrine, notwithstanding the repeated refrain that disclaimer applies only where the applicant's disavowal of claim scope must be clear and unambiguous. In some cases, for example, disclaimer may be inferred from a single statement concerning the prior art.[134] But in other cases, what appeared to be a disclaiming statement could be disregarded in view of the prosecution history as a whole.[135] In another case, the disclaimer turned on the interpretation of the "i.e." in the applicant's remarks to mean "for example" to find no prosecution disclaimer.[136] And in some cases, this Court has carefully analyzed whether the applicant's discussion of the prior art raised a point of distinction rather than one of claim scope, while in

---

[134] *See Vehicle IP, LLC v. Gen. Motors Corp.*, 306 F. App'x 574, 578 (Fed. Cir. 2009) (inferring that "[i]f the location of the mobile unit must be determined independently of the notification coordinate, then the notification coordinate necessarily must provide an absolute location," to find prosecution disclaimer).

[135] *See Elbex Video, Ltd. v. Sensormatic Elecs. Corp.*, 508 F.3d 1366, 1372-1373 (Fed. Cir. 2007) (court found that though "read in isolation, the statement in the prosecution history" could be a disclaimer, the whole file history must be considered when evaluating prosecution disclaimer).

[136] *See Toshiba Corp. v. Imation Corp.*, 681 F.3d 1358, 1370 (Fed. Cir. 2012).

others, statements about the contents of the prior art alone were sufficient constitute disclaimer.[137] Indeed, current disclaimer law resulted in multiple different constructions of the very term at issue here.[138]

E. Applicants Did Not Disclaim the Accused "Ring Oscillator"

When examined in the context of the full prosecution history, the disclaimers adopted in this case evidence the problems inherent in a court relying on statements in the prosecution history to try to assess the intentions of an applicant an and examiner, particularly when the record is incomplete. For example, here there were numerous telephone discussions between the Examiner and Applicants over a period of several years. The calls are not transcribed, and there are no examiner summaries. Thus, the record is at least partially obscured and unknown.

Across the span of several rejections and several years, Applicants and the Examiner had an on-going dialogue. A review of the entire dialogue, an arduous

---

[137] *Compare Shire Dev., LLC v. Watson Pharm., Inc.*, 787 F.3d 1359, 1366 (Fed. Cir. 2015) (no disclaimer when "Shire carefully characterized the prior art as not having separate matrices but never actually stated that the claimed invention does have separate matrices."), *with Krippelz v. Ford Motor Co.*, 667 F.3d 1261, 1266-67 (Fed. Cir. 2012) (applicant argued that prior art reference lacked a "light bulb at or near the focal point of a reflector" and therefore "disclaimed lamps lacking these limitations, and the limitations therefore became part of the properly construed claims.")

[138] Appx6718-6719.

task fraught with the risk of hindsight revision, reveals a different result than was reached here. Before Magar was cited, Applicants and Examiner had reached agreement that the independent claims, as amended, overcame Sheets, thus calling into question the imposition of an additional disclaimer based on Sheets. The same is true with regard to Magar. Notably, "entire" was not added to the claims until after the statements that have been found to be disclaimers were made. By applying additional negative limitations beyond the amendments to the claims, the Court simply supplanted its judgment for that of the Examiner.

Even more importantly, however, every statement found to be a disclaimer in this case related solely to Applicants' unsuccessful efforts to obtain broad independent claims that required only the microprocessor and a single clock (the "Optimal CPU Clock Scheme") over single-clock prior art. At the suggestion of the Examiner, Applicants agreed to restrict the independent claims to require all of the elements of Claim 6 at issue in this case, including a second external clock. Put in plain language, the Examiner effectively said: "Look, let's agree to disagree about whether you are entitled to the broad claim with just a microprocessor and an integrated clock, but I will give you what you are asking for (the claimed clock on the same circuit with the CPU) as long as your further limit the claim—and add another clock and the other elements of Figure 17 from the dependent claims!" Notably, the Examiner never required Applicants to amend to add anything like the

two disclaimers found in this case. Applicants and the Examiner reached equilibrium that none the prior art disclosed the combination of the Optimal CPU Clock Scheme with the Asynchronous/Synchronous CPU—and thus, the Examiner granted Claim 6. Under the Supreme Court disclaimer jurisprudence, the only disclaimer that occurred in this case was the disclaimer of a claim only requiring two elements.

But this Court decided that Applicants disclaimed subject matter and imposed two negative limitations. The consequences of adopting these limitations could not have been anticipated. The negative limitations have now been applied find that the very structure disclosed as the "entire oscillator" in the specification that the applicant was arguing should be allowed over the prior art is non-infringing. But this is a nonsensical result.

Nothing in the file history suggests that Applicants intended to disclaim the disclosed oscillator. Indeed, Applicants explained in each excerpt of the file history relied on to adopt a negative limitation why the "variable speed ring oscillator" in the specification and the claims was allowable over the cited references. For both references, Applicants essentially explained the basic characteristics of a ring oscillator and the advantages of having it on the same chip as the CPU. The claimed combination meant that the oscillator could supply the

clock for the CPU without the need for an external crystal and did not require a command input. It is that simple.

But now, based on these negative limitations, a variable speed ring oscillator has been summarily adjudicated to be non-infringing. Appellees have used the negative limitations to argue that the addition of the PLL takes the variable speed ring oscillator outside the scope of the claims. This result is wrong.

F. Adopting Limitations Based on Statements Other Than Amendments Has Other Unintended Results

Unintended, unpredictable, and inconsistent results are not the only problem with the manner in which disclaimer law has evolved. As occurred here, courts now can and do adopt claim limitations that are not grounded in the language of the specification, but instead solely in attorney argument in the prosecution history. Here, the Court added limitations that include the phrases "command input" and "fixed by an external oscillator" that appear nowhere in the claims or specification. This has effectively multiplied the proceedings here as no consensus on the purported scope of the disclaimers has been reached even now.

Notably, in summary judgment briefing, both parties again argued different positions on the Sheets and Magar notwithstanding the fact that these issues had already purportedly been resolved. And there can be no question that meaning of "fixed" and "command input" remain in dispute. In addition to arguing the scope of Sheets in their summary judgment brief, Appellees even submitted dictionary

definitions for both "command" and "input."[139] With regard to "fixed," as described in detail below, Appellees advocate a definition of fixed that takes it well outside of the scope of the purported disclaimer.

G. Simplifying Disclaimer Law Promotes the Public Notice Function of the Patent

Adhering to Supreme Court precedent, which (1) allows only *clear and unequivocal* statements of disavowal *based only on* amendments to the claims, amendments to the specification, or statements that "I do not claim X," (and not merely attorney arguments as to why prior art was distinguishable) and (2) analyzes disclaimer in the context of infringement analysis, would restore needed predictability to patent claim interpretation. Such adherence would eliminate the need for courts to reinterpret what the patent examiner "intended" or "would have intended" or "likely intended" in allowing claims. Given the inconsistent results and the manner in which courts struggle to divine what an applicant and examiner intended by various statements—as is demonstrated by how multiple tribunals have given different interpretations of the patent at issue here—the expansive view of disclaimer adopted in more recent lower and appellate court decisional law actually undermines the public notice nature of the patent that these cases claim to promote.

[139] Appx5476-5479; *see also* Appx5253.

Under current non-Supreme Court jurisprudence, courts add limitations to claims through a process of tasseography based on an after-the-fact analysis of an applicant's statements contained in an incomplete record of what transpired (such as the interview summaries). It is unreasonable to think that the public could reasonably rely on such statements. Indeed, there is no evidence in this case that Appellees referred to, relied on or even considered the file history in making any decisions when they designed the products at issue, or chose to continue using them after they learned of the patent. Instead, as is most often the case today, disclaimer law is used by infringers *after the fact* to try to avoid what would otherwise be a clear case of infringement (as is the case here), forcing courts and counsel to spend countless hours delving into vague, lengthy, and cumbersome patent prosecution history.

Efforts to re-interpret claims after issuance inevitably leads to, and here did lead to, exactly what the Supreme Court in *Goodyear* cautioned against: use of "*correspondence* between the applicant … and the Commissioner … *to enlarge, diminish, or vary the language of a patent* afterwards issued.[140] Thus, Appellants assert that disclaimer is a judicially created doctrine that is best served by a return to its roots within the parameters authorized by the Supreme Court. This is not to

[140] *Goodyear Dental*, 102 U.S. at 227 (emphasis added).

say that, consistent with *Markman*, the prosecution history should not be consulted for the *meaning* of claims terms, but absent an "I disclaim x" statement, amendments to the patent claims should control the inquiry on disclaimer.

Analyzing the facts of this case with reference to Supreme Court disclaimer jurisprudence, Appellants contend the finding of disclaimer should be vacated. Contrary to this Court's prior claim construction, the oscillator—as claimed, and without any negative limitations—was actually allowed, albeit in combination of the other elements of the Asynchronous/Synchronous CPU. Appellants respectfully request that this Court: revise its prior claim; vacate the summary judgment order; and reverse and remand the case for further proceedings.

III. THE DISTRICT COURT IMPROPERLY IMPORTED FUNCTIONAL LIMITATIONS INTO APPARATUS CLAIMS

That the disclaimer found in this case creates much confusion about the scope of the claims is perhaps best evidenced by the district court's Summary Judgment Order.

The asserted claims are apparatus claims. Infringement of an apparatus claim is found when an accused device meets the structural limitations of the claims. Notwithstanding the well-settled maxim that "apparatus claims cover what a device is, not what a device does," the district court applied this Court's claim construction to impart functional requirements on the entire oscillator and granted

summary judgment of non-infringement based on the purported behavior of the accused devices.[141] The district court's approach was wrong as a matter of law.

The district court's opinion granting summary judgment reveals that the Court misapplied this Court's claim construction and imposed (at Appellees' urging) a restrictive functional requirement not found in the claims, nor within the ambit of any disclaimer by the patentee. Indeed, the district court's summary judgment order confirms this: "The record shows that, ***unlike the free-running oscillators*** described in the patent, the accused oscillators are situated within PLLs that hold their frequencies ***essentially steady*** until they are changed by a new command input."[142] Neither the claims nor the prior claim construction of this Court requires that the oscillators ***operate*** in a "free running" mode but that is exactly the basis under which the district granted summary judgment of non-infringement.

The district court imparted functional limitations akin to those found in a method claim rather than applying the structural limitations of the claims as construed. Yet, despite the district court's apparent finding that the accused ring oscillators are variable, and do, in fact, vary in response to PVT parameters, it

---

[141] *Hewlett–Packard Co. v. Bausch & Lomb, Inc*., 909 F.2d 1464, 1468 (Fed.Cir.1990)

[142] Appx8 (emphasis added).

decided that they did not vary ***enough***: "these minor fluctuations do not constitute the changes in clock frequency contemplated by the Federal Circuit's claim construction."[143]

But this Court's prior claim construction did not speak in degrees, nor impose a minimum fluctuation. The district court's confusion may stem from the Appellees' assertion that assessment of whether the "entire oscillator" is variable should be evaluated by comparison of its output during operation with the output of a crystal oscillator.[144] Accepting that premise, the district court concluded that there is "no reason to consider any minor frequency variations occurring within *a locked PLL* to be a change in clock frequency identified in the Federal Circuit's claim construction."[145] In other words, the court concluded that because the reference crystal is fixed, the PLL system's output is fixed, and thus the accused products did not infringe. This conclusion is based on a fundamental misapplication of this Court's claim construction, and by amalgamating the two disclaimers in a manner unsupported by the record. The district court's decision to

[143] Appx5; *see also* Appx6 ("PLLs inhibit frequency changes of *any significance*" (emphasis added)); Appx7 ("changes resulting in frequency resulting from operational parameters are all but imperceptible").

[144] *See* Appx5 ("The record shows that, within a PLL, the accused oscillators operate at frequencies comparably stable to those of crystal oscillators.").

[145] Appx5.

impart the separate "fixed by any external crystal" limitation into its "command input" analysis improperly ignores the context giving rise to the separate disclaimers and occurred even though the district court chose not to discuss or analyze the "fixed crystal" portion of the disclaimer.[146]

On the basis of the District Court's misapplication of the claim construction order alone, this case should be reversed and remanded.

IV. SUMMARY JUDGMENT SHOULD BE VACATED BECAUSE THE APPELLEES FAILED TO NEGATE AN ESSENTIAL ELEMENT AND THERE ARE DISPUTED QUESTIONS OF FACT

A. Appellees' Motion Was Directed Only to Whether A PLL Met The "Entire Oscillator" Limitation

When moving for summary judgment, "the moving party must produce evidence negating an essential element of the nonmoving party's claim or defense or show that the nonmoving party does not have enough evidence of an essential element to carry its ultimate burden of persuasion at trial," and if the "moving party fails to carry its initial burden of production," "the nonmoving party may defeat the motion for summary judgment without producing anything."[147] Here, Appellees misdirect their arguments, failing to address whether the accused

[146] *See* Appx8; *see also Massachusetts Inst. of Tech. v. Shire Pharm*., Inc., 839 F.3d 1111, 1119-1124 (Fed. Cir. 2016) (separately analyzing each term).

[147] *See, e.g.*, *Nissan Fire & Marine Ins. Co. v. Fritz Companies, Inc.*, 210 F.3d 1099, 1102–03 (9th Cir. 2000).

structure meets this Court's construction, and have therefore failed to meet their initial burden.

In this case, the "entire oscillator" has been interpreted to be "an oscillator located entirely on the same semiconductor substrate as the central processing unit that does not require a command input to change the clock frequency and whose frequency is not fixed by any external crystal."[148] In accord with this definition, Appellants identified the ring oscillator in the accused products as the structure that met this apparatus claim limitation. Appellees did not challenge that the ring oscillator meets the structural limitations of this claim element. Instead, Appellees argued ***only*** that the PLL fails to meet this limitation. For example, Appellees argue (1) that the PLL produces a fixed frequency output (*e.g.*, Appx5259-5260); (2) that the PLL requires a command input (*e.g.*, Appx5264-5266); and (3) that the PLL uses a crystal as a reference point (*e.g.*, Appx5274-5275).

Appellants contend that these arguments about the PLL fail as a matter of law. But nonetheless, there is no question that the district court's decision improperly accepted this strawman argument, and for this reason as well, the district court's decision should be reversed.

[148] *Tech. Properties Ltd. LLC v. Huawei Techs. Co.*, 849 F.3d 1349, 1360 (Fed. Cir. 2017).

B. The Addition of PLL Circuitry Does Not Defeat Infringement

The addition of circuitry beyond the identified ring oscillators does not defeat infringement, nor does it alter the ring oscillator's structure or characteristics. Appellees' arguments that the PLL system's response to and attempts to mitigate the inherent variations in the ring oscillators defeats infringement contradicts long-standing precedent—one cannot defeat infringement by merely adding unclaimed elements:

> It is fundamental that ***one cannot avoid infringement merely by adding elements if each element recited in the claims is found in the accused device***. For example, a pencil structurally infringing a patent claim would not become noninfringing when incorporated into a complex machine that limits or controls what the pencil can write. Neither would infringement be negated simply because the patentee failed to contemplate use of the pencil in that environment.[149]

Like the pencil in *A.B. Dick*, the accused ring oscillator is incorporated into a more complex arrangement (the PLL system)[150] that the district court found limits or controls what the ring oscillator can output.[151] Also like the pencil, the ring oscillator is the structurally infringing element to which additional structure is

---

[149] *A.B. Dick Co. v. Burroughs Corp*., 713 F.2d 700, 703 (Fed. Cir. 1983), *cert. denied*, 464 U.S. 1042 (1984) (citing *Temco Elec. Motor Co. v. Apco Mfg. Co*., 275 U.S. 319, 328 (1928)) (emphasis added).

[150] *E.g.*, Appx5298 ¶14 ("The operation of the PLLs in the Accused Samsung Processors is complex.").

[151] Appx4, Appx6.

added in effort to control its output (i.e., what the pencil can write). Importantly, the structure and fundamental characteristics of the ring oscillator remain unchanged even when the PLL system is added around it—notwithstanding the addition of the PLL, the ring oscillator remains a variable frequency oscillator whose output frequency will vary based on a processing parameter (how it is manufactured) or an operating parameter (temperature or voltage), as claimed.[152] Appellees acknowledge this.[153] The district court thus incorrectly found that the addition of the PLL renders the accused products non-infringing.

This Court recently clarified in *Skedco, Inc. v. Strategic Operations, Inc*., that a structural component of a device will not lose its inherent qualities just because another structural component adjusts its output.[154] In *Skedco*, the asserted claims required, amongst other structures, a pump, a valve, and a controller. The district court held that the accused device, which combined a pump, valve, and associated controls within a single housing, did not infringe because the recited structures had to be separate. This Court overturned that holding and clarified that

---

[152] Appx6541.

[153] *See, e.g.*, Appx5259 ("Like a cruise control, ***the PLL compensates for any PVT effects*** on its transistors and circuitry ...." (emphasis added)).

[154] *Skedco, Inc. v. Strategic Operations, Inc.*, 685 F. App'x 956, 959 (Fed. Cir. 2017).

placing the pump within a housing did not alter the pump's status as a pump: "We see no reason why a device that moves fluid cannot contain another device that regulates flow within it. A pump does not cease moving fluid—i.e., being a 'pump'—just because an internal valve adjusts fluid flow."[155] Like the pump in *Skedco*, the ring oscillator does not cease being a variable-speed oscillator whose clock speed varies depends on processing or operational parameters just because the PLL circuitry (like the valve in *Skedco*) adjusts the oscillator's frequency afterwards.

C. The District Court Improperly Rejected *A.B. Dick*

The district court's rejection of the long-standing *A.B. Dick* precedent appears to flow from its injection of the functional limitations and quantitative requirements for the assessing variability based on PVT variations, and its conclusion that the accused devices would need to be altered to satisfy them.[156] But that holding is divorced from the structural requirement that the relevant oscillators vary together with the CPU because they are located on the same semiconductor substrate.[157] These structural features are unchanged by the addition of PLL circuitry.

---

[155] *Id*.

[156] *See* Appx7.

[157] *TPL*, 849 F.3d at 1360.

The cases the district court cited are inapposite and do not change the outcome here. The facts and holding in *Outside the Box Innovations, LLC v. Travel Caddy, Inc.*,[158] cited by the district court, are inapplicable here, and do not usurp the application of *A.B. Dick*. In *Outside the Box*, the claims required a "flexible fabric" panel, but the accused devices included a fabric-covered plywood panel—materially different from the claimed structure.[159] There is no material change in the structure of the claimed oscillators in the present case; the PLL may adjust the frequency of the on-chip ring oscillator, but the on-chip ring oscillator's structure is unchanged.

Nor is any change or alteration to the accused products necessary to demonstrate infringement—they infringe "as is" because they meet the structural limitations of the asserted apparatus claims. For this reason, *High Tech Med. Instrumentation, Inc. v. New Image Indus., Inc.*,[160] is also inapplicable.

D. There Are, At a Minimum, Disputed Questions of Fact That Preclude Entry of Summary Judgment

Even if the disclaimers are applied as written, the record provides sufficient evidence that the accused oscillators satisfy the "entire oscillator" limitation, and

---

[158] 695 F.3d 1285, 1305 (Fed. Cir. 2012) (per curiam).

[159] *Id*.

[160] 49 F.3d 1551, 1555 (Fed. Cir. 1995) ("The original and intended operating configuration of the device must be altered—by loosening the set screws—in order for the camera to rotate.").

thus, there are disputed questions of fact that should have precluded summary judgment.

In fact, a jury determined as much in the prior *HTC v. TPL* trial, where representative products with the same representative architectures were found to infringe the asserted claims, based on testimony from Qualcomm and Texas Instruments corporate witnesses, in addition to both of the parties' experts, all of whom testified that the on-chip variable speed ring oscillators generate the system clock for the CPU.[161]

1. The Ring Oscillators in the Accused Devices Do Not "Require a Command Input"

Under this Court's prior construction, the entire oscillator "does not ***require*** a command input to change the clock frequency."[162] Appellants' expert opined that the accused VCOs meet this limitation:

> It is my opinion that the ring oscillators in the accused products meet the Federal Circuit's definition of "entire oscillator." The frequencies generated by an on-chip ring oscillator in a PLL System necessarily depend on fabrication and operation parameters, such as process, voltage and temperature. Indeed, the PLL is employed in

[161] Appx1815, Appx1824-1825 (on-chip ring oscillators generate the system clock; the added PLL uses an external crystal as a reference signal). Applicability of the *HTC* findings are underscored because some of the same processors are at issue here. *Compare* Appx1764 (MSM7x30 in accused HTC devices), *with* Appx5027 (MSM7230 in accused Huawei devices).

[162] *TPL*, 849 F.3d at 1359-60 (emphasis added).

> the accused products to manage the VCO because of its inherent variability to PVT—that is its purpose. Notwithstanding the presence of added PLL circuitry, the ring oscillator (VCO) will oscillate as soon as power is applied, and its frequencies vary based on fabrication and operational parameters….[163]

This salient feature of ring oscillators is a matter of physics. Temperature changes and voltage changes will change the frequency of the on-chip oscillators—no command inputs required.[164]

Even Appellees' motion and expert declaration tacitly admit that the clock frequencies of the identified ring oscillators (VCOs) vary in response to PVT. Appellees assert that the PLL circuitry acts like a cruise control, constantly stepping in to counteract the ring oscillators' natural responses to PVT: "Like a cruise control, ***the PLL compensates for any PVT effects*** on its transistors and circuitry …."[165] Appellees therefore admit that ***PVT effects change the ring oscillators' clock frequencies***. Put another way, if the frequency did not vary based on PVT, there would be no need for constant intervention by the PLL "cruise control" to correct them.

---

[163] Appx6539 ¶33.

[164] *See* Appx6541 ¶38; *see also* Appx5017 (citing *HTC* trial testimony); Appx5254 ("the PLL feedback loop compensates for PVT variations").

[165] Appx5259 (emphasis added).

2. The Ring Oscillators in the Accused Devices Are Not "Fixed By Any External Crystal"

The district court chose not to analyze or address Appellees' second basis for seeking summary judgment of non-infringement (Appellees' assertion that the accused devices are fixed by an external crystal). Out of abundance of caution and to avoid assertions of waiver, Appellants nonetheless address this issue.

Whether the VCOs in the accused devices are "fixed by an external crystal" at least remains a disputed question of material fact for at least three reasons: (1) the output frequencies of the VCOs are not fixed at all; (2) properly applying the "Magar" disclaimer, the VCOs are not fixed because their frequencies are not generated by any external crystal; and (3) the external crystal merely provides a reference frequency that is used by a comparator in the PLL circuitry, and thus does not "fix" the VCO output frequency.[166]

First, by definition and according to their structure, the output frequencies of the accused VCOs are "variable," and are therefore not fixed (much less by an external crystal).[167] Indeed, the variability of the accused VCOs necessitates stabilization by the PLL circuitry. But even then, there are ***still*** variations in VCO

[166] Appellants maintain that there was no disclaimer, but infringement may be shown even if disclaimer applies.

[167] Appx5328-5329, Appx5333; Appx6540-6542.

output frequency.[168] In addition, even Appellees' expert and declarant admit that the accused VCOs are designed to operate within a range of frequencies and are set to one of a number of frequencies.[169] Each of these reasons raise disputed questions of fact that preclude summary determination.

Second, the VCOs are not "fixed" under a proper application of this Court's claim construction, because "fixed by an external crystal" in the context of the alleged disclaimer means that the external crystal actually ***generates*** the clock signal. But that is not the case in the accused products (where the ring oscillator generates the clock signal, and the external crystal merely provides a reference frequency that is never communicated to the ring oscillator).[170]

The accused VCOs are "variable-speed ring oscillators." The accused VCOs generate the oscillations that form the system clock; they do not depend on an off-chip crystal to oscillate.[171] Appellants' expert opined that the accused VCOs are

---

[168] *E.g.*, Appx1781 (addressing variations in *HTC* matter).

[169] Appx5299-5300 ¶¶16, 18-19; *see also, e.g.*, Appx5592, Appx5600; Appx5694-5695; Appx5535-5538; Appx5551-5557; Appx5331 ¶50.

[170] *See TPL*, 849 F.3d at 1358-59 (citing Appx2092-2093, Appx2101-2102); Appx2101 ("Magar's clock generator relies on an external crystal … to oscillate…").

[171] *See* Appx6539 ¶¶33-34, Appx6540-6541 ¶¶36-37; Appx1815, Appx1820-1821, Appx1824-1825.

responsible for the generating the clock frequencies for the CPUs in the accused devices.[172] Appellees' expert admitted this as well.[173]

Third, it is undisputed that the off-chip crystal in any accused device merely furnishes a reference signal that is used by a comparator (or "phase detector") in the PLL to assess the difference between the frequency generated by the VCO and the reference.[174] The off-chip crystal is not even an input to the VCO—nor can it be, because the off-chip crystal outputs a frequency, while the VCO's input must be a voltage.[175] This evidence that the off-chip crystal only provides a reference for use by circuitry in the PLL, and the crystal's signal ***never even reaches*** the VCO, at minimum raises a dispute of fact as to whether the reference crystal could "fix" the VCO.

In sum, because the off-chip reference crystal does not generate the clock for the on-chip ring oscillator, the reference crystal is not connected to the ring oscillator, and the signal from the reference crystal is never communicated to the ring oscillator, there is ample evidence from which a reasonable jury could

---

[172] Appx6539-6541 ¶¶34, 38; *see also* Appx1818.

[173] *See, e.g.,* Appx5327 ¶43 ("The output of the VCO is used to drive the CPU."); Appx5298-5300 ¶¶16-17.

[174] *See* Appx6539-6541 ¶¶34-36; Appx5299 ¶16; Appx5326-5327; Appx1824-1825.

[175] *E.g.*, Appx6539-6541 ¶¶34-38.

conclude that the accused on-chip ring oscillators (VCOs) are not "fixed by any external crystal."

V. CONCLUSION

For these reasons, Appellants respectfully request that this Court vacate the district court's summary judgment order and entry of judgment and remand this case for further proceedings.

Respectfully Submitted,

Dated: April 23, 2018

/s/ Denise De Mory

Denise M. De Mory (SBN 168076)
Henry C. Bunsow (SBN 60707)
Aaron R. Hand (SBN 245755)
Lauren N. Robinson (SBN 255028)
BUNSOW DE MORY LLP
701 El Camino Real
Redwood City, CA 94063
Telephone: (650) 351-7248
Facsimile: (415) 426-4744

ddemory@bdiplaw.com
hbunsow@bdiplaw.com
ahand@bdiplaw.com
lrobinson@bdiplaw.com

Counsel for Appellants/Plaintiffs,
TECHNOLOGY PROPERTIES LIMITED LLC,
PHOENIX DIGITAL SOLUTIONS LLC,
PATRIOT SCIENTIFIC CORPORATION

ADDENDUM

TABLE OF CONTENTS TO ADDENDUM

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

TECHNOLOGY PROPERTIES LIMITED LLC, et al., Plaintiffs, v. HUAWEI TECHNOLOGIES CO., LTD., et al., Defendants.	**JUDGMENT** Case No. 12-cv-03865-VC
TECHNOLOGY PROPERTIES LIMITED LLC, et al., Plaintiffs, v. ZTE CORPORATION, et al., Defendants.	Case No. 12-cv-03876-VC
TECHNOLOGY PROPERTIES LIMITED LLC, et al., Plaintiffs, v. SAMSUNG ELECTRONICS CO., LTD., et al., Defendants.	Case No. 12-cv-03877-VC

TECHNOLOGY PROPERTIES LIMITED LLC, et al., Plaintiffs, v. LG ELECTRONICS, INC., et al., Defendants.	Case No. 12-cv-03880-VC
TECHNOLOGY PROPERTIES LIMITED LLC, et al., Plaintiffs, v. NINTENDO CO., LTD., et al., Defendants.	Case No. 12-cv-03881-VC

The Court, having granted the defendants' motions for summary judgment, now enters judgment in favor of the defendants and against the plaintiffs in each of the above five cases. The Clerk of Court is directed to close these cases.

IT IS SO ORDERED.

Dated: December 13, 2017

VINCE CHHABRIA
United States District Judge

UNITED STATES DISTRICT COURT

NORTHERN DISTRICT OF CALIFORNIA

TECHNOLOGY PROPERTIES LIMITED LLC, et al.,

Plaintiffs,

v.

HUAWEI TECHNOLOGIES CO., LTD., et al.,

Defendants.

ORDER GRANTING MOTIONS FOR SUMMARY JUDGMENT

Case No. 12-cv-03865-VC

Re: Dkt. No. 139

TECHNOLOGY PROPERTIES LIMITED LLC, et al.,

Plaintiffs,

v.

ZTE CORPORATION, et al.,

Defendants.

Case No. 12-cv-03876-VC

Re: Dkt. No. 143

TECHNOLOGY PROPERTIES LIMITED LLC, et al.,

Plaintiffs,

v.

SAMSUNG ELECTRONICS CO., LTD., et al.,

Defendants.

Case No. 12-cv-03877-VC

Re: Dkt. No. 140

TECHNOLOGY PROPERTIES LIMITED LLC, et al., Plaintiffs, v. LG ELECTRONICS, INC., et al., Defendants.	Case No. 12-cv-03880-VC Re: Dkt. No. 157
TECHNOLOGY PROPERTIES LIMITED LLC, et al., Plaintiffs, v. NINTENDO CO., LTD., et al., Defendants.	Case No. 12-cv-03881-VC Re: Dkt. No. 141

The defendants' motions for summary judgment of non-infringement are granted.

The plaintiffs ("TPL") stipulated to non-infringement under this Court's prior construction of the phrase "an entire oscillator disposed upon said integrated circuit substrate" as used in the asserted claims of Patent No. 5,809,336. The Federal Circuit then made a "minor modification" to that claim construction, holding that the proper construction of the disputed claim term is: "an oscillator located entirely on the same semiconductor substrate as the central processing unit that does not require a command input to change the clock frequency and whose frequency is not fixed by any external crystal." *Tech. Props. Ltd. LLC v. Huawei Techs. Co.*, 849 F.3d 1349, 1360 (Fed. Cir. 2017). In doing so, the Federal Circuit noted that its change to the prior construction "likely does not affect the outcome in this case." *Id.* The Federal Circuit's prediction was correct.

The parties do not dispute that the oscillators within the accused products operate as part of "phase-locked loop" systems ("PLLs"). The parties agree that, in practice, these PLLs limit the frequencies at which the oscillators at issue oscillate. *See, e.g.*, Decl. of Dr. Vivek Subramanian at 21, Dkt. No. 139-3; Decl. of Dr. Vojin Oklobdzija at 9-10, Dkt. No. 142-1. The parties also essentially agree on how PLLs work: PLLs use a reference frequency, generally

provided by an off-chip crystal oscillator, along with a programmable divisor to set the frequency of the on-chip system clock. As a result, within a functioning PLL, the frequency at which the on-chip oscillator oscillates is a multiple of the off-chip reference frequency. *See* Subramanian Decl. at 17-20; Oklobdzija Decl. at 10; *id.* at 14 ("A PLL proportionally tracks the reference frequency as closely as possible").

TPL argues that, even within the PLL, the accused oscillators infringe because they experience frequency variations resulting from process, voltage, and temperature parameters for which the PLL must correct. *See* TPL Opp'n Br. at 23-26, 30-31, Dkt. No. 142. Because the oscillators are inherently responsive to these parameters, TPL contends, the accused oscillators do not "require a command input to change the clock frequency." But, assuming that some small frequency variations occur while the PLL is operating, these minor fluctuations do not constitute the changes in clock frequency contemplated by the Federal Circuit's claim construction.

The record shows that, within a PLL, the accused oscillators operate at frequencies comparably stable to those of crystal oscillators. *See* Subramanian Decl. at 28-33; Decl. of Erik Fuehrer, Ex. 6 at 1217-26, 1480-83, Dkt. No. 138-16; *see also* TPL Opp'n Br. at 24 ("At most, Defendants' testing shows that PLLs stabilize the output of on-chip oscillators . . . and that those stabilized outputs are roughly similar in stability to a frequency output by a hypothetical crystal."). TPL characterizes crystal oscillators as "fixed." *See* TPL Opp'n Br. at 2 ("A clock signal generated from a crystal is a fixed-frequency signal that does not meaningfully vary based on environmental conditions."); Fuehrer Decl., Ex. 2 at 4, Dkt. No. 139-6, ("Crystals are by design fixed-frequency devices whose oscillation speed is designed to be tightly controlled and to vary minimally due to variations in manufacturing, operating voltage and temperature"). There is thus no reason to consider any minor frequency variations occurring within a locked PLL to be the changes in clock frequency identified in the Federal Circuit's claim construction. *See Tech. Props. Ltd.*, 849 F.3d at 1360.[1]

[1] There is also no reason to think that the Federal Circuit intended to refer to differences between the maximum frequency capabilities of one processor versus another in crafting the limitation

The record further shows that the frequency of the on-chip oscillator within the PLL will remain stable, in the sense discussed above, unless and until it is changed by a command input, namely, a change to the crystal that sets the reference frequency or to the value of a programmable divisor within the PLL. *See* Subramanian Decl. at 20; Decl. of Marzio Pedrali-Noy at 3-4, Dkt. No. 138-12; Decl. of Dr. Jaegon Lee at 6, 11, Dkt. No. 138-10. TPL has provided no evidence to the contrary, nor has it provided a definition of "command input" that would exclude inputs of these kinds. *Cf.* Oklobdzija Decl. at 12 (pointing only to the oscillator's "fundamental characteristics . . . determined by physics and nature" as support for the notion that no command input is required to change the clock frequency).

It's worth noting that, because PLLs inhibit frequency changes of any significance in the absence of a command input, PLLs prevent the oscillators in the accused devices from acting in the advantageous manner touted in the relevant part of the patent and recognized by the Federal Circuit. The proposed benefit of locating the claimed oscillator on the same substrate as the CPU is that the clock and the CPU can "automatically vary together," without requiring a command input to change the clock frequency. *Tech. Props. Ltd.*, 849 F.3d at 1360 (citation omitted); Fuehrer Decl., Ex. 3 at 7, Dkt. No. 139-7 ("[T]he operational speed of the microprocessor and ring oscillator clock are designed to vary similarly as a function of variation in temperature, processing and other parameters affecting circuit performance"); *see also* Oklobdzija Decl. at 7. The effectively simultaneous, corresponding changes in the frequencies of the clock and CPU allow the CPU to run "at the maximum frequency possible, but never too fast" given the process, voltage, and temperature conditions affecting the CPU. '336 Patent at 17:1-2, Dkt. No. 139-5; *see also* Fuehrer Decl., Ex. 3 at 7-9. Rather than allow the frequency of the oscillator to vary freely with process, voltage, and temperature parameters as in the claimed

---

regarding command inputs and changes in clock frequency. Therefore, to the extent TPL contends that the practice of "binning," in which manufacturers sort processors based on their performance capabilities, is evidence that the accused oscillators can change frequency as a result of fabrication process parameters, not just command inputs, the argument is not persuasive.

invention, the PLL controls the frequency at which its component oscillator oscillates so that its frequency does not track changes in these parameters. And, as mentioned, the undisputed evidence shows that the PLL does so very effectively, such that any changes in frequency resulting from operational parameters are all but imperceptible.

In its papers and through its experts, TPL makes an alternative argument (although counsel for TPL seemed – wisely – to disavow it at oral argument). The argument is that what matters is not how the accused oscillators operate within a PLL, but whether the accused oscillators in isolation meet all the claim limitations. *See, e.g.*, Oklobdzija Decl. at 13 (stating that the relevant testing to determine infringement "would need to measure the [voltage-controlled oscillator's] frequencies with PLL circuitry disabled so that the VCO frequency changes in response to temperature were not masked by PLL intervention."). But the accused oscillators don't operate in isolation in the accused devices, they operate within the tightly controlled framework of the PLL. Given the claim limitations at issue and the construction provided by the Federal Circuit, TPL cannot defeat the defendants' summary judgment motions simply by asserting that the accused devices hypothetically could infringe if altered. In other words, that the accused products all situate the on-chip oscillator within a PLL matters for purposes of determining whether those products infringe, because the PLLs affect how the on-chip oscillator's frequency is determined; the PLL circuitry is not simply an extra element added on to an infringing device. *See Outside the Box Innovations, LLC v. Travel Caddy, Inc.*, 695 F.3d 1285, 1305 (Fed. Cir. 2012) (per curiam) (concluding that the addition of plywood to a fabric panel was not merely a feature added on to an infringing device but a "material change" such that the accused product did not infringe the claimed "flexible fabric . . . panel"); *High Tech Med. Instrumentation, Inc. v. New Image Indus., Inc.*, 49 F.3d 1551, 1555 (Fed. Cir. 1995) (holding that a patentee was unlikely to succeed in proving infringement where, to infringe, "[t]he original and intended operating configuration of the device must be altered" by loosening screws fixing the accused camera in place); *see also Accent Packaging, Inc. v. Leggett & Platt, Inc.*, 707 F.3d 1318, 1327 (Fed. Cir. 2013). The question is not whether the accused oscillators

could infringe in theory, but whether there is any dispute about whether they do in fact.

In sum, TPL has not put forth evidence sufficient to raise a question about whether the oscillators in the accused products require a command input to change the frequencies at which they oscillate. The record shows that, unlike the free-running oscillators described in the patent, the accused oscillators are situated within PLLs that hold their frequencies effectively steady until they are changed by a command input. Because it is clear that the accused devices require a command input to change the clock frequency, they do not meet "each and every limitation" of the asserted claims. *Cross Med. Prod., Inc. v. Medtronic Sofamor Danek, Inc.*, 424 F.3d 1293, 1310 (Fed. Cir. 2005). Summary judgment for the defendants is appropriate, and there's no need to discuss whether the accused oscillators are "fixed by any external crystal," although it seems likely that TPL would lose on that question as well. *Tech. Props. Ltd.*, 849 F.3d at 1360.

IT IS SO ORDERED.

Dated: December 13, 2017

VINCE CHHABRIA
United States District Judge

US005809336A

United States Patent [19]

Moore et al.

[11] Patent Number: 5,809,336

[45] Date of Patent: Sep. 15, 1998

[54] **HIGH PERFORMANCE MICROPROCESSOR HAVING VARIABLE SPEED SYSTEM CLOCK**

[75] Inventors: **Charles H. Moore**, Woodside; **Russell H. Fish, III**, Mt. View, both of Calif.

[73] Assignee: **Patriot Scientific Corporation**, San Diego, Calif.

[21] Appl. No.: **484,918**

[22] Filed: **Jun. 7, 1995**

Related U.S. Application Data

[62] Division of Ser. No. 389,334, Aug. 3, 1989, Pat. No. 5,440,749.

[51] **Int. Cl.**[6] **G06F 1/04**

[52] **U.S. Cl.** **395/845**

[58] **Field of Search** 395/500, 551, 395/555, 845

[56] **References Cited**

U.S. PATENT DOCUMENTS

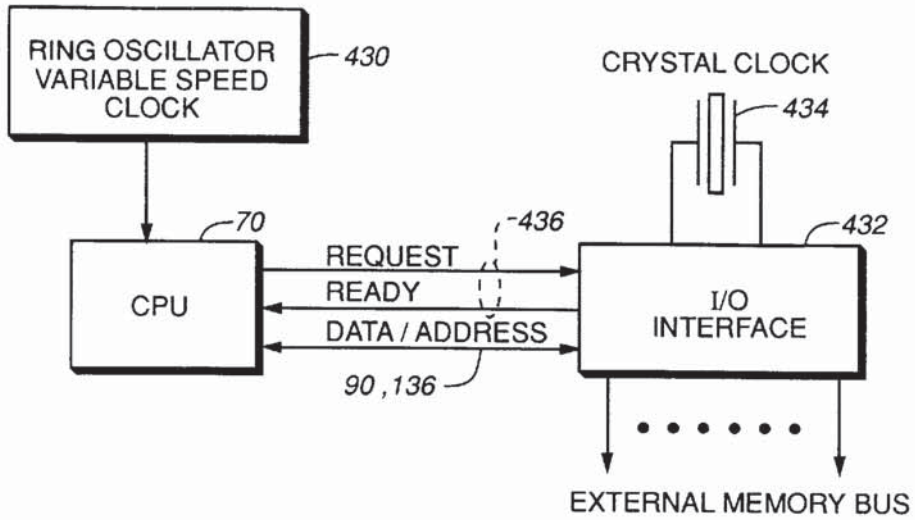
3,967,104	6/1976	Brantingham	364/709.09
3,980,993	9/1976	Bredart et al.	395/550
4,003,028	1/1977	Bennett et al.	395/742
4,042,972	8/1977	Gruner et al.	395/389
4,050,096	9/1977	Bennett	395/494
4,112,490	9/1978	Pohlman et al.	395/287
4,315,308	2/1982	Jackson	395/853
4,338,675	7/1982	Palmer	364/748
4,398,265	8/1983	Puhl et al.	395/882
4,453,229	6/1984	Schaire	395/250
4,503,500	3/1985	Magan	395/800
4,539,655	9/1985	Trussell et al.	395/280
4,553,201	11/1985	Pollack	395/183.22
4,627,082	12/1986	Pelgrom et al.	377/63
4,670,837	6/1987	Sheets	395/550
4,680,698	7/1987	Edwards et al.	395/800
4,761,763	8/1988	Hicks	395/286
5,414,862	5/1995	Suzuki et al.	395/750

Primary Examiner—David Y. Eng
Attorney, Agent, or Firm—Cooley Godward LLP

[57] **ABSTRACT**

A high performance, low cost microprocessor system having a variable speed system clock is disclosed herein. The microprocessor system includes an integrated circuit having a central processing unit and a ring oscillator variable speed system clock for clocking the microprocessor. The central processing unit and ring oscillator variable speed system clock each include a plurality of electronic devices of like type, which allows the central processing unit to operate at a variable processing frequency dependent upon a variable speed of the ring oscillator variable speed system clock. The microprocessor system may also include an input/output interface connected to exchange coupling control signals, address and data with the central processing unit. The input/output interface is independently clocked by a second clock connected thereto.

10 Claims, 19 Drawing Sheets



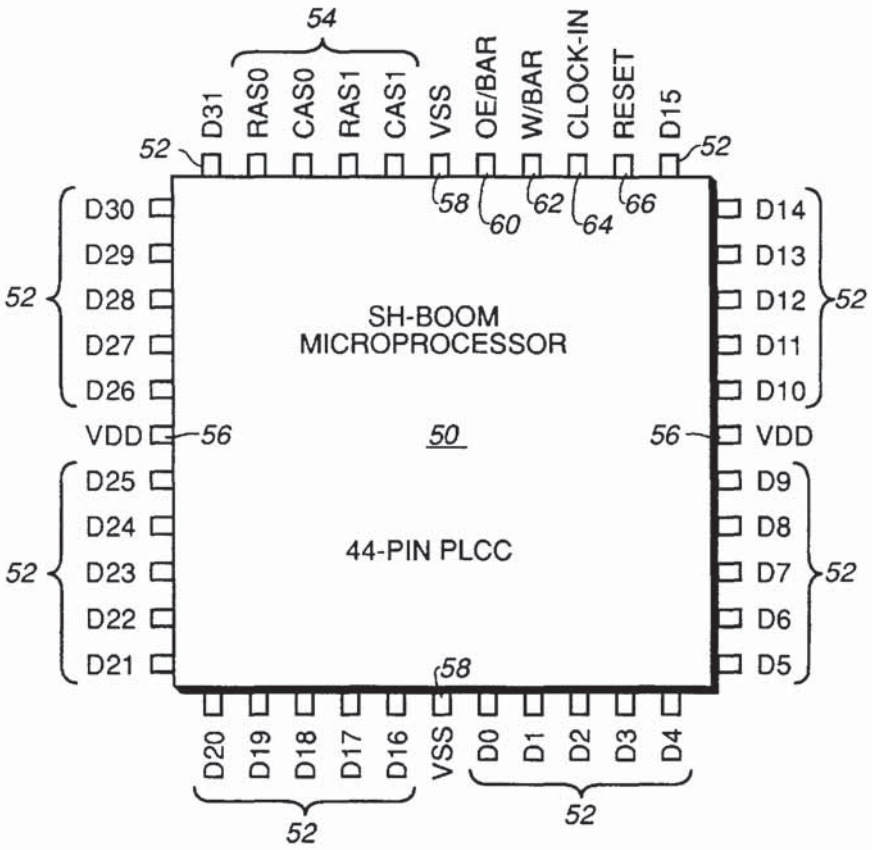

54
D31
RAS0
CAS0
RAS1
CAS1
VSS
OE/BAR
W/BAR
CLOCK-IN
RESET
D15
52
58
60
62
64
66
D30
D29
D28
D27
D26
VDD
56
SH-BOOM
MICROPROCESSOR
50
44-PIN PLCC
D14
D13
D12
D11
D10
D25
D24
D23
D22
D21
D9
D8
D7
D6
D5
D20
D19
D18
D17
D16
D0
D1
D2
D3
D4

FIG._1

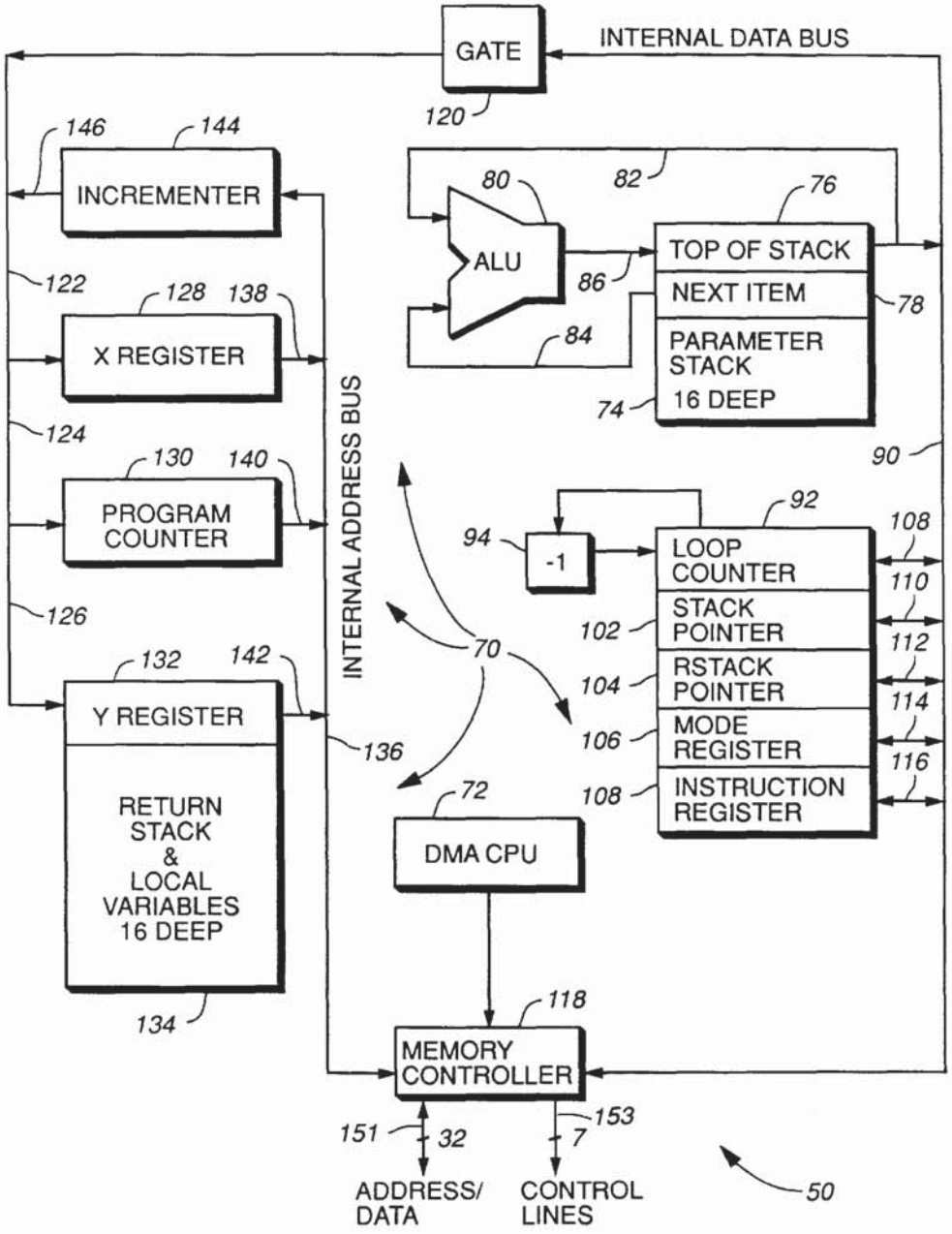

GATE
INTERNAL DATA BUS
120
146
144
INCREMENTER
80
82
76
ALU
TOP OF STACK
NEXT ITEM
PARAMETER STACK
16 DEEP
86
84
78
74
122
128
138
X REGISTER
124
130
140
PROGRAM COUNTER
INTERNAL ADDRESS BUS
90
92
94
-1
LOOP COUNTER
STACK POINTER
RSTACK POINTER
MODE REGISTER
INSTRUCTION REGISTER
108
110
112
114
116
102
104
106
108
70
126
132
142
Y REGISTER
136
RETURN STACK & LOCAL VARIABLES 16 DEEP
134
72
DMA CPU
118
MEMORY CONTROLLER
151
32
153
7
ADDRESS/ DATA
CONTROL LINES
50

FIG._2

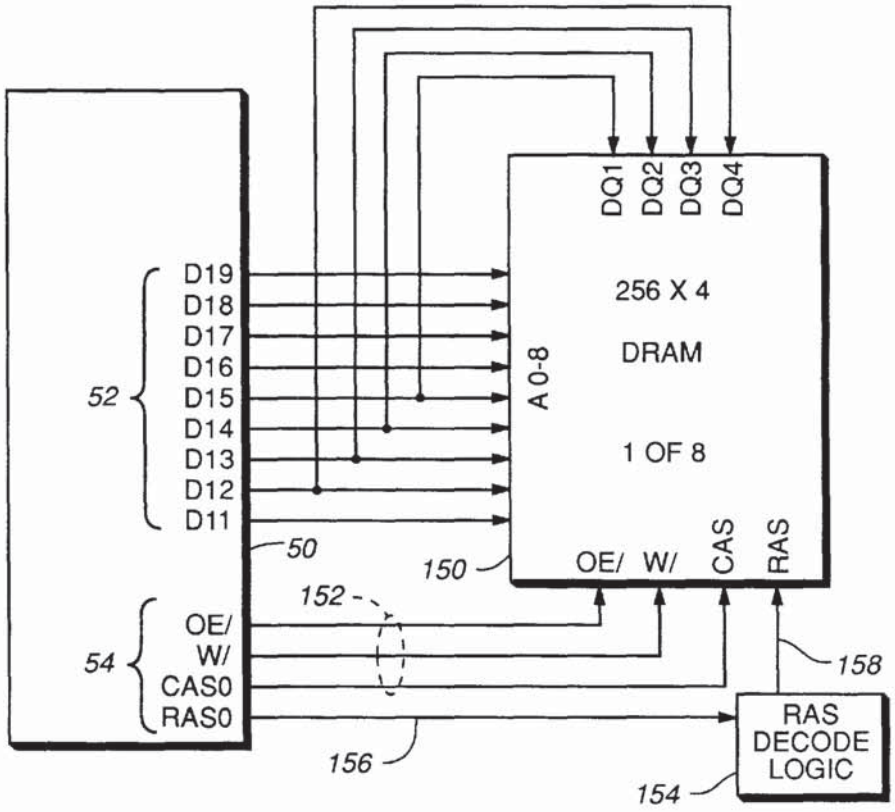

DQ1
DQ2
DQ3
DQ4
256 X 4
DRAM
1 OF 8
A 0-8
OE/
W/
CAS
RAS
D19
D18
D17
D16
D15
D14
D13
D12
D11
52
50
150
152
OE/
W/
CAS0
RAS0
54
156
158
RAS
DECODE
LOGIC
154

FIG._3

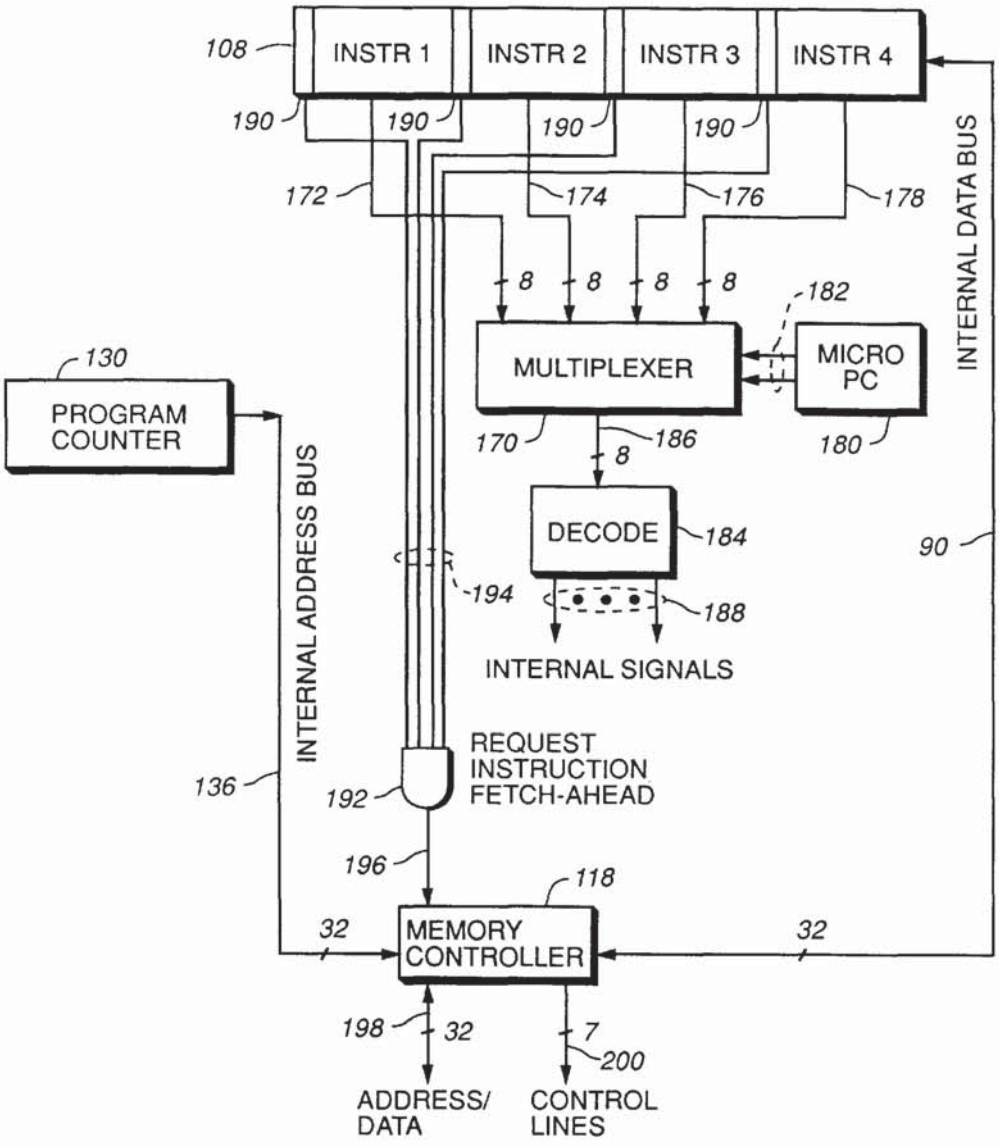

108
INSTR 1
INSTR 2
INSTR 3
INSTR 4
190
190
190
190
172
174
176
178
8
8
8
8
182
MULTIPLEXER
MICRO PC
180
130
PROGRAM COUNTER
170
186
8
DECODE
184
194
188
INTERNAL SIGNALS
INTERNAL ADDRESS BUS
INTERNAL DATA BUS
90
136
192
REQUEST INSTRUCTION FETCH-AHEAD
196
118
32
MEMORY CONTROLLER
32
198
32
7
200
ADDRESS/ DATA
CONTROL LINES

FIG._4

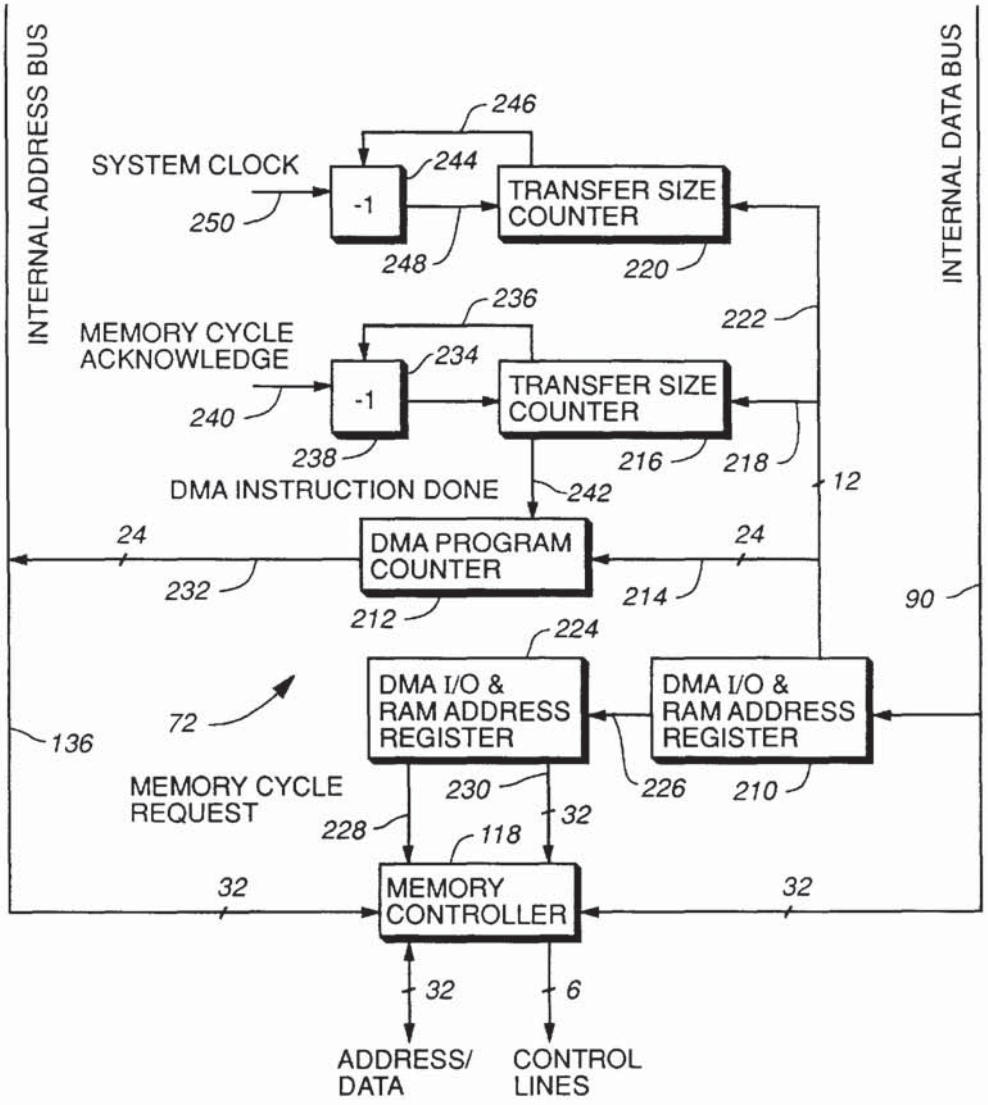

INTERNAL ADDRESS BUS
INTERNAL DATA BUS
SYSTEM CLOCK
250
246
244
-1
248
TRANSFER SIZE COUNTER
220
MEMORY CYCLE ACKNOWLEDGE
240
236
234
-1
238
TRANSFER SIZE COUNTER
216
218
222
12
DMA INSTRUCTION DONE
242
24
DMA PROGRAM COUNTER
232
212
214
24
90
224
DMA I/O & RAM ADDRESS REGISTER
DMA I/O & RAM ADDRESS REGISTER
72
136
MEMORY CYCLE REQUEST
230
226
210
228
32
118
32
MEMORY CONTROLLER
32
32
32
6
ADDRESS/ DATA
CONTROL LINES

FIG._5

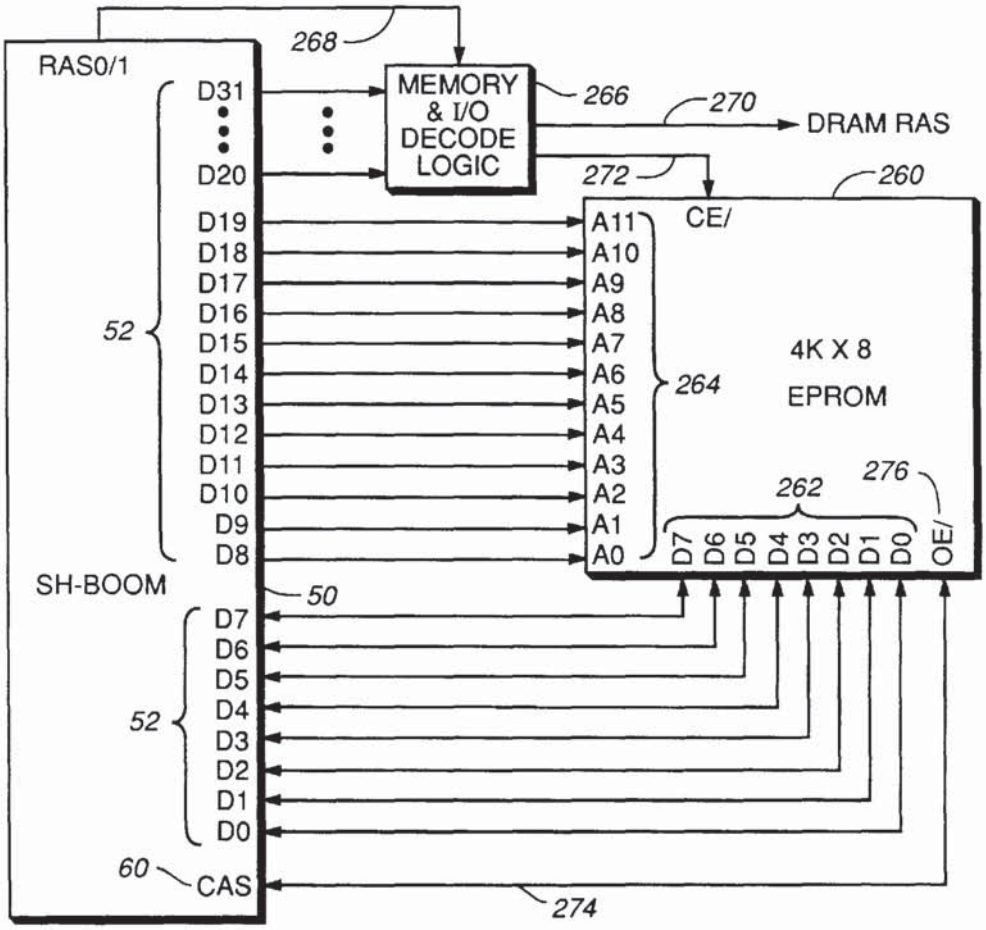

268
RAS0/1
D31
D20
MEMORY & I/O DECODE LOGIC
266
270
DRAM RAS
272
260
CE/
D19
D18
D17
D16
D15
D14
D13
D12
D11
D10
D9
D8
52
A11
A10
A9
A8
A7
A6
A5
A4
A3
A2
A1
A0
264
4K X 8
EPROM
262
276
D7
D6
D5
D4
D3
D2
D1
D0
OE/
SH-BOOM
50
52
60
CAS
274

FIG._6

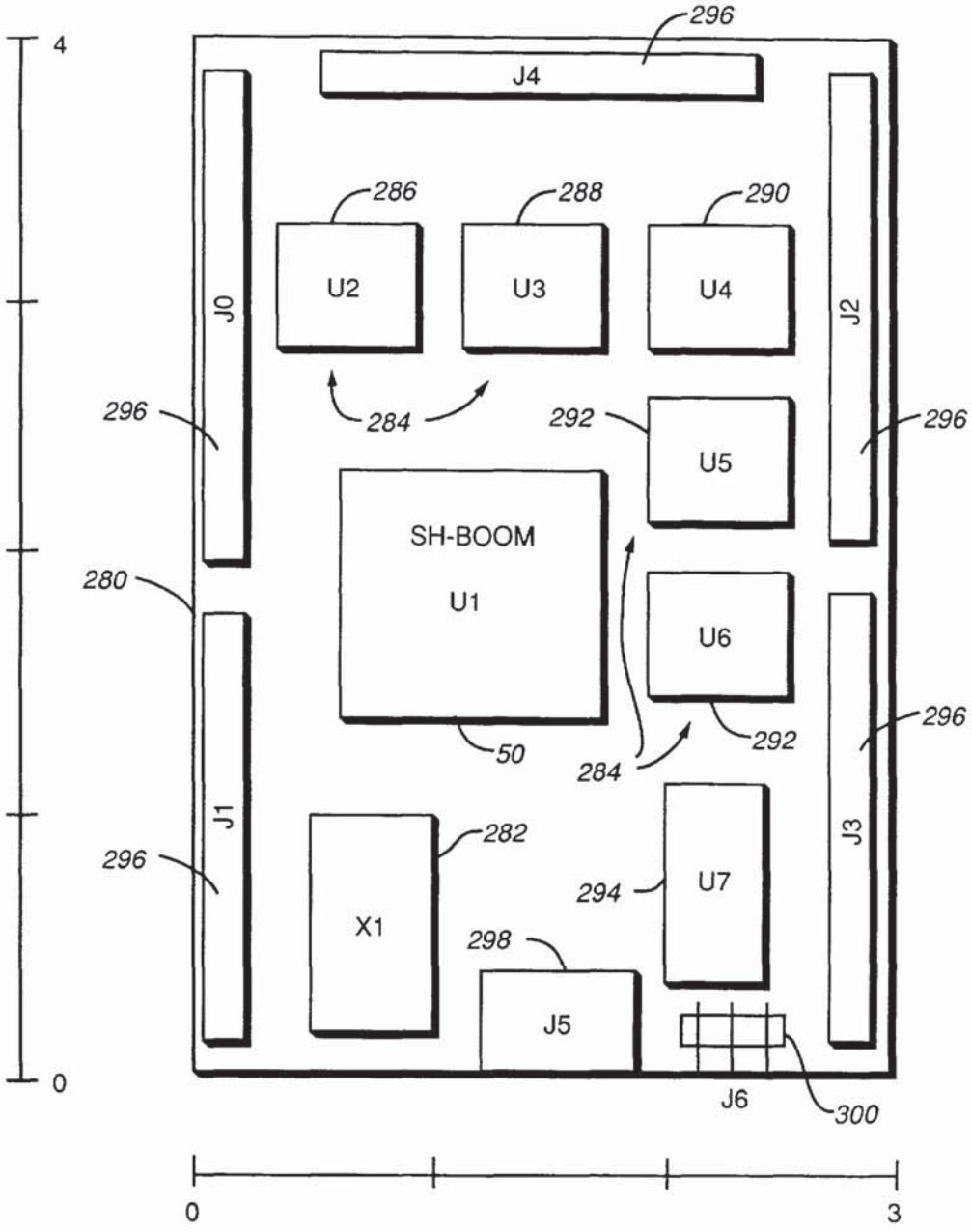

296
J4
286
288
290
U2
U3
U4
J0
J2
284
296
292
296
U5
SH-BOOM
U1
280
U6
296
50
284
292
J1
J3
296
282
294
U7
X1
298
J5
J6
300
4
0
0
3

FIG._7

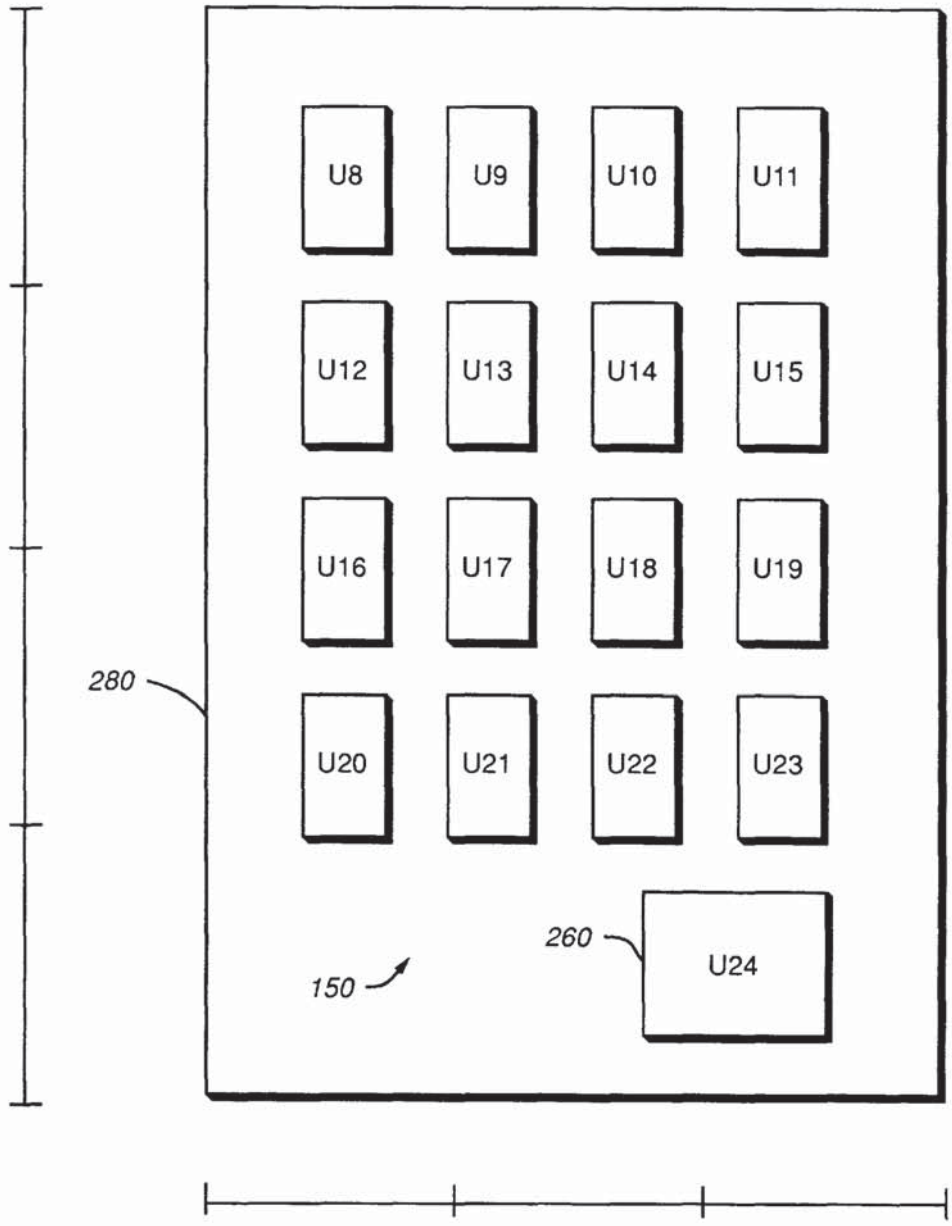

U8
U9
U10
U11
U12
U13
U14
U15
U16
U17
U18
U19
U20
U21
U22
U23
U24
280
260
150

FIG._8

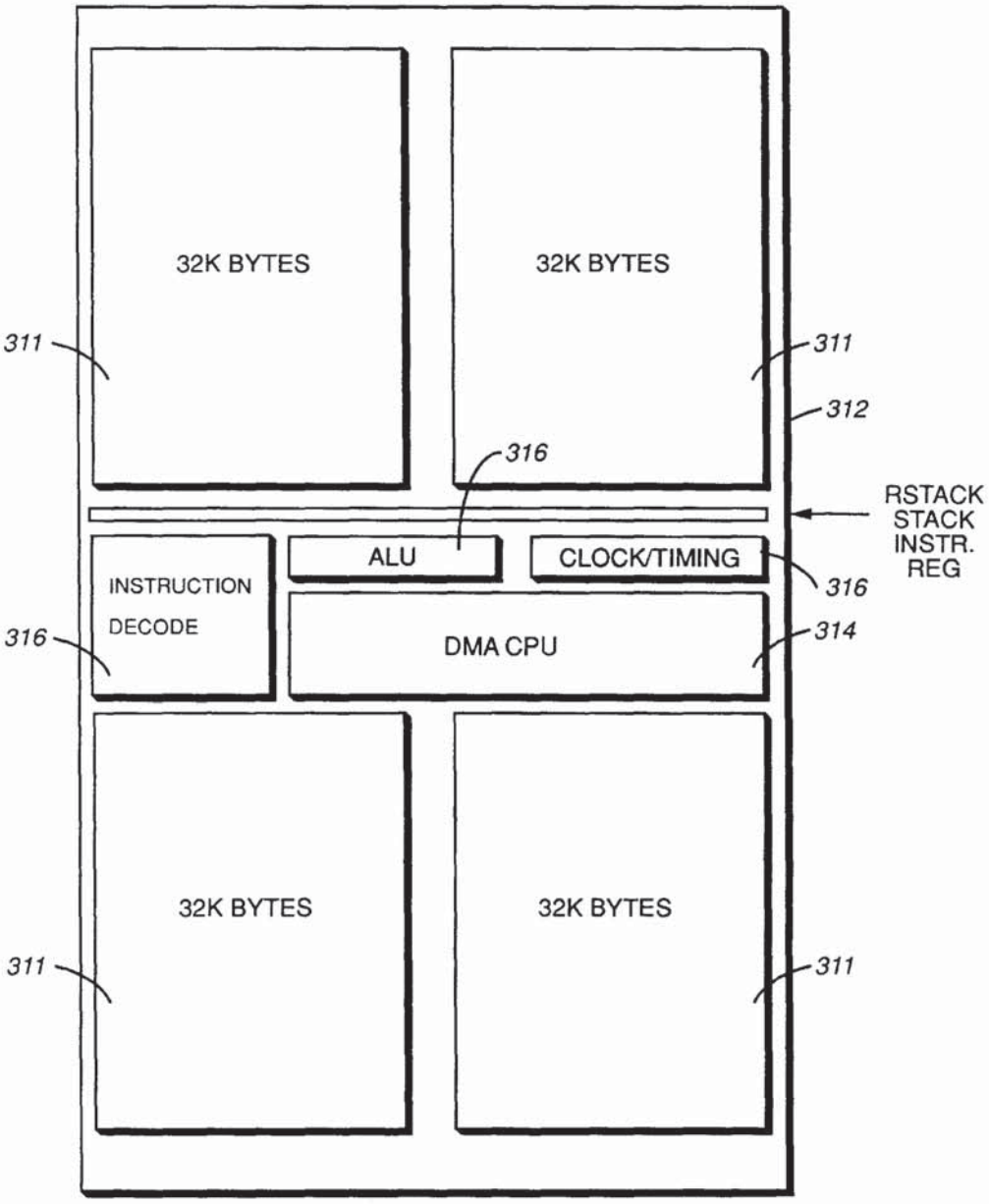

32K BYTES
32K BYTES
311
311
312
316
RSTACK
STACK
INSTR.
REG
ALU
CLOCK/TIMING
316
INSTRUCTION
DECODE
316
DMA CPU
314
32K BYTES
32K BYTES
311
311

FIG._9

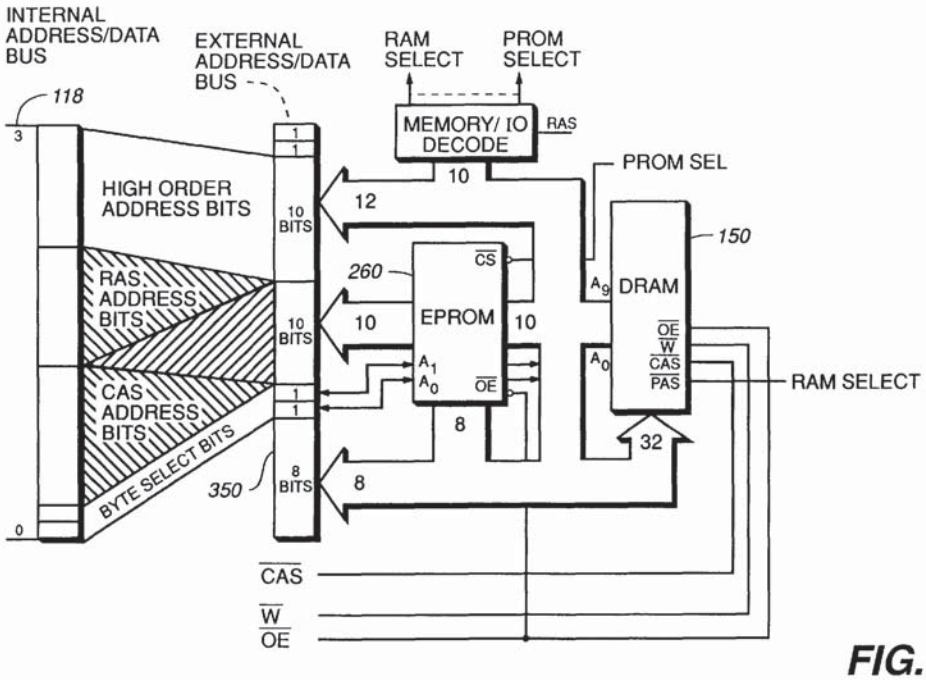
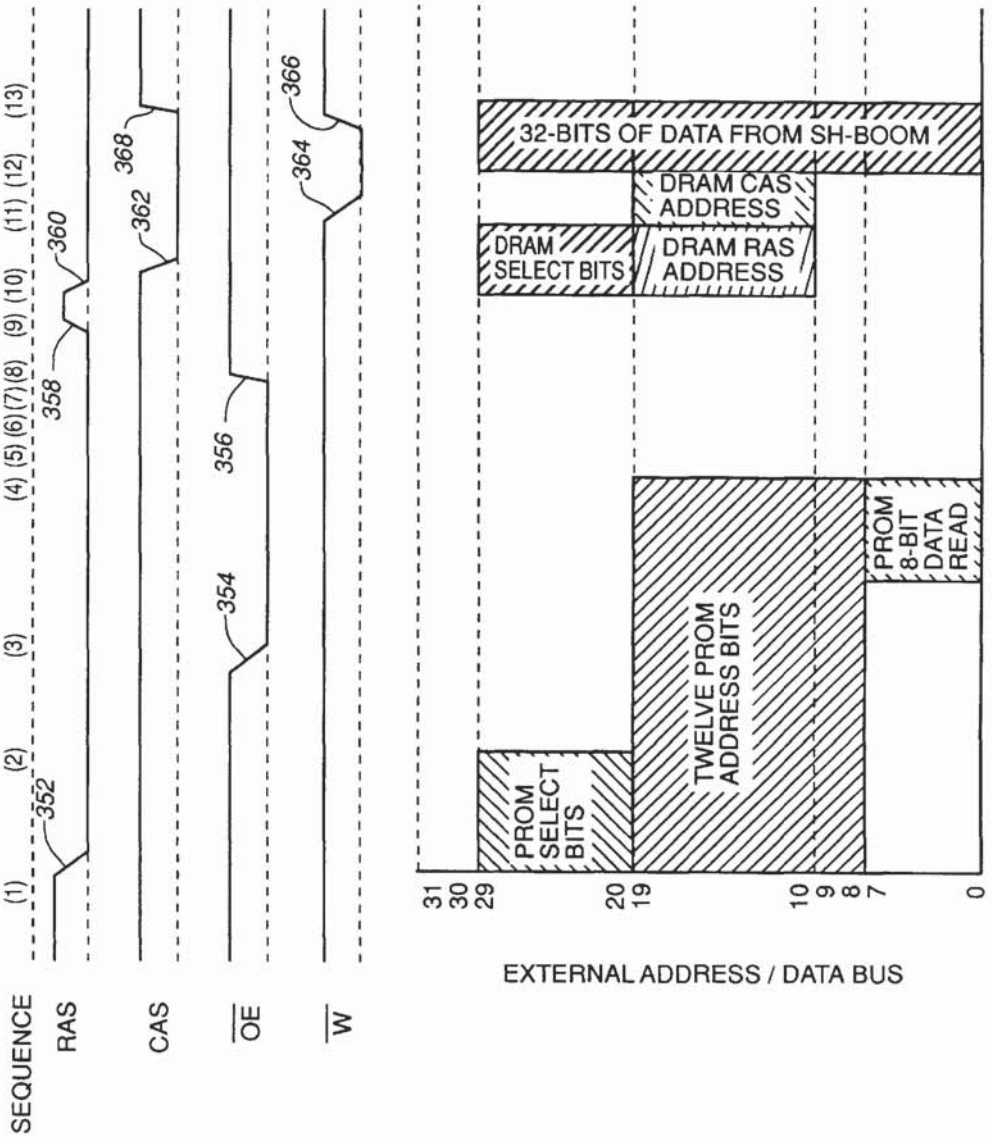

INTERNAL ADDRESS/DATA BUS
EXTERNAL ADDRESS/DATA BUS
RAM SELECT
PROM SELECT
118
3
0
HIGH ORDER ADDRESS BITS
RAS ADDRESS BITS
CAS ADDRESS BITS
BYTE SELECT BITS
1
1
10 BITS
10 BITS
1
1
8 BITS
350
MEMORY/ IO DECODE
RAS
10
12
PROM SEL
260
CS
EPROM
10
10
A1
A0
OE
8
8
150
A9
DRAM
A0
OE
W
CAS
PAS
RAM SELECT
32
CAS
W
OE

FIG._10

FIG._11


SEQUENCE
RAS
CAS
OE
W
(1)
(2)
(3)
(4)
(5)
(6)
(7)
(8)
(9)
(10)
(11)
(12)
(13)
352
354
356
358
360
362
364
366
368
32-BITS OF DATA FROM SH-BOOM
DRAM CAS ADDRESS
DRAM SELECT BITS
DRAM RAS ADDRESS
PROM SELECT BITS
TWELVE PROM ADDRESS BITS
PROM 8-BIT DATA READ
31
30
29
20
19
10
9
8
7
0
EXTERNAL ADDRESS / DATA BUS

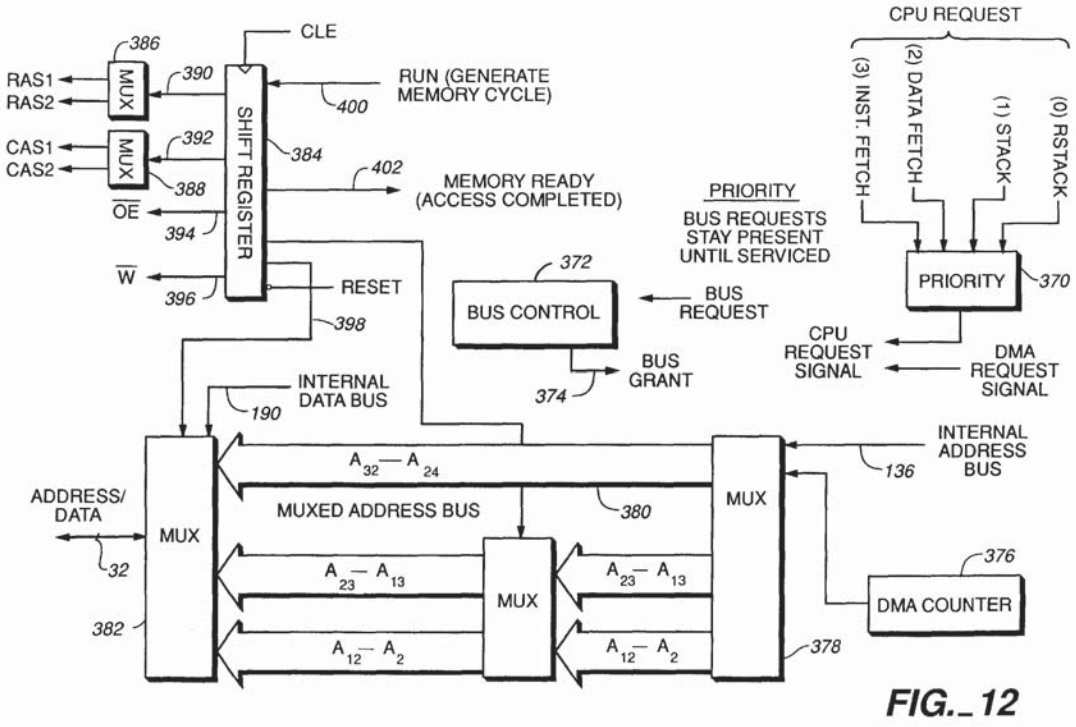

CPU REQUEST
(3) INST. FETCH
(2) DATA FETCH
(1) STACK
(0) RSTACK
CLE
386
RAS1
RAS2
MUX
390
SHIFT REGISTER
RUN (GENERATE MEMORY CYCLE)
400
CAS1
CAS2
MUX
392
384
402
MEMORY READY (ACCESS COMPLETED)
388
OE
394
W
396
RESET
398
PRIORITY
BUS REQUESTS STAY PRESENT UNTIL SERVICED
372
BUS CONTROL
BUS REQUEST
PRIORITY
370
CPU REQUEST SIGNAL
DMA REQUEST SIGNAL
BUS GRANT
374
INTERNAL DATA BUS
190
INTERNAL ADDRESS BUS
136
A32 — A24
MUXED ADDRESS BUS
380
ADDRESS/ DATA
MUX
32
A23 — A13
MUX
A23 — A13
MUX
376
DMA COUNTER
382
A12 — A2
A12 — A2
378

FIG._12

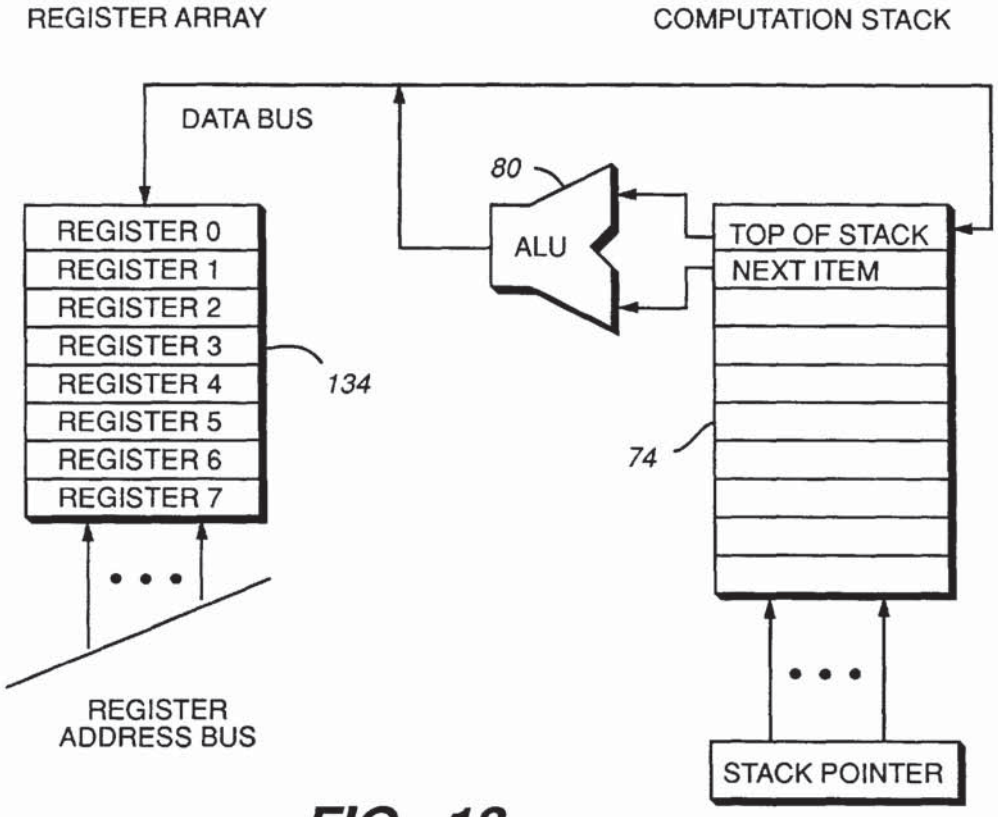

REGISTER ARRAY
COMPUTATION STACK
DATA BUS
80
ALU
REGISTER 0
REGISTER 1
REGISTER 2
REGISTER 3
REGISTER 4
REGISTER 5
REGISTER 6
REGISTER 7
134
TOP OF STACK
NEXT ITEM
74
REGISTER
ADDRESS BUS
STACK POINTER

FIG._13

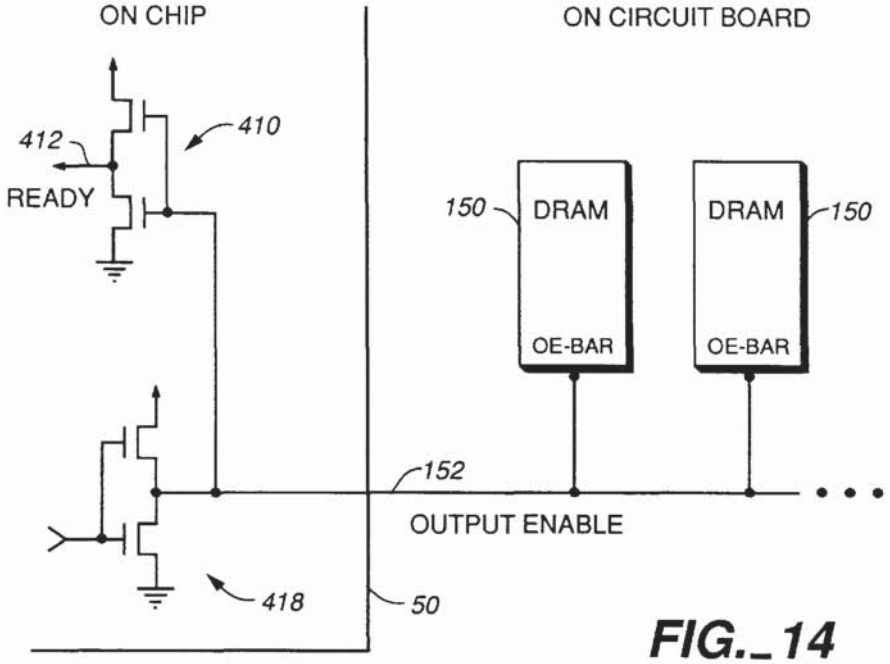

ON CHIP
ON CIRCUIT BOARD
412
410
READY
150
DRAM
OE-BAR
DRAM
150
OE-BAR
152
OUTPUT ENABLE
418
50

FIG._14

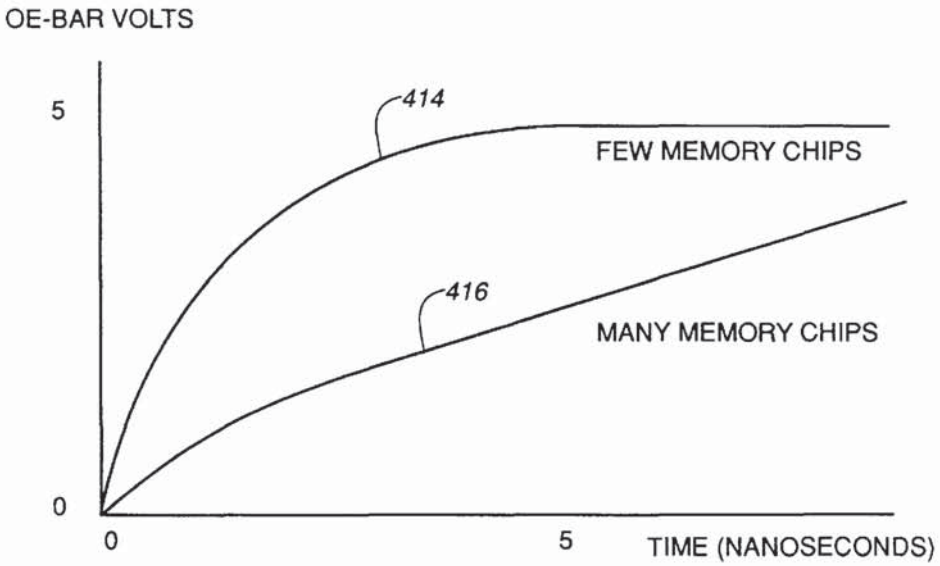

OE-BAR VOLTS
5
0
414
FEW MEMORY CHIPS
416
MANY MEMORY CHIPS
0
5
TIME (NANOSECONDS)

FIG._15

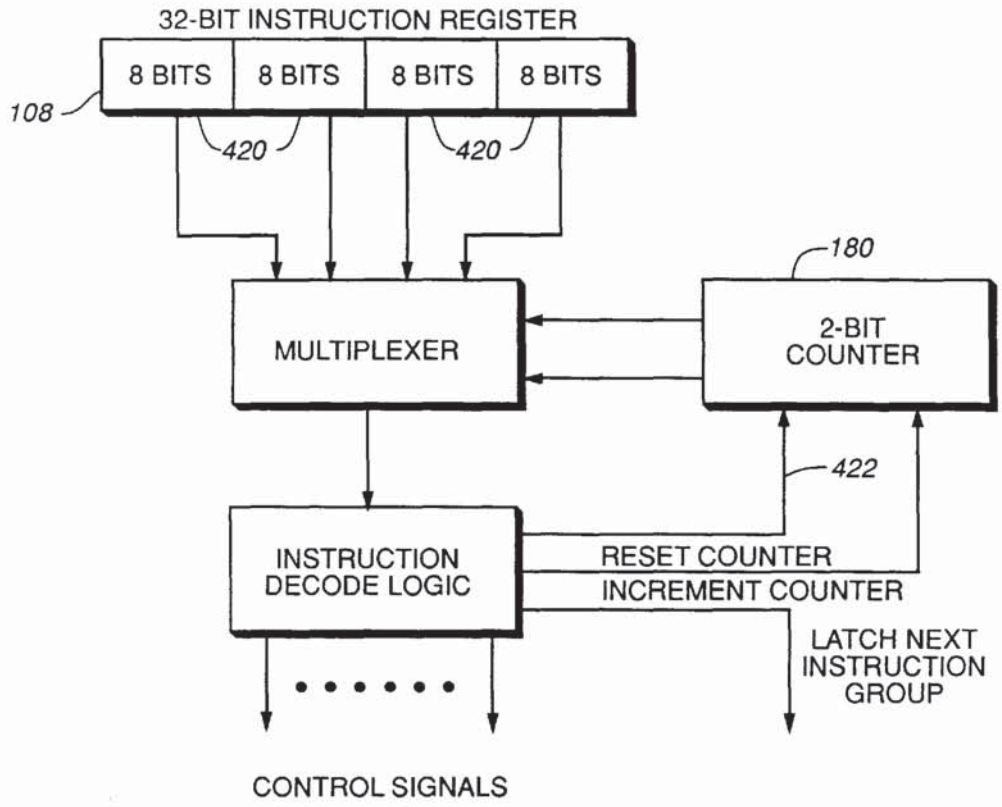

32-BIT INSTRUCTION REGISTER
8 BITS
8 BITS
8 BITS
8 BITS
108
420
420
180
MULTIPLEXER
2-BIT COUNTER
422
INSTRUCTION DECODE LOGIC
RESET COUNTER
INCREMENT COUNTER
LATCH NEXT INSTRUCTION GROUP
CONTROL SIGNALS

FIG._16

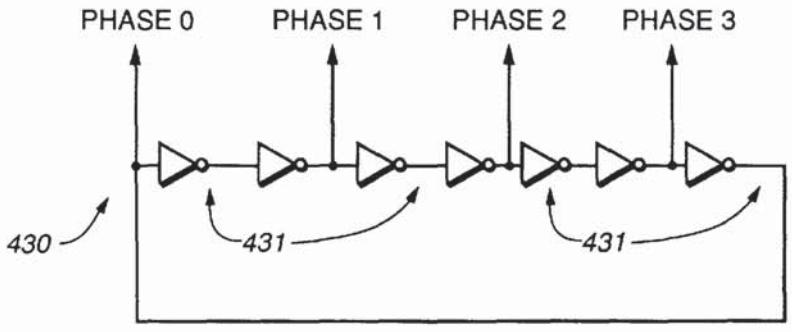

PHASE 0
PHASE 1
PHASE 2
PHASE 3
430
431
431

FIG._18

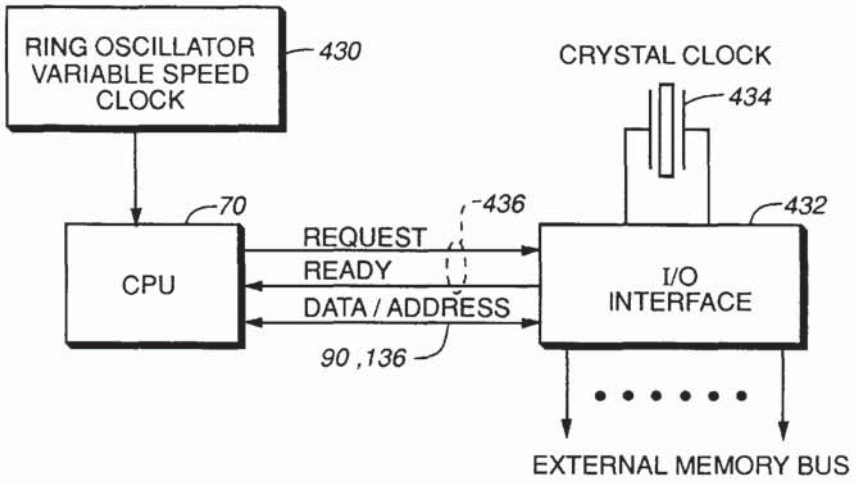

RING OSCILLATOR
VARIABLE SPEED
CLOCK
430
CRYSTAL CLOCK
434
70
436
432
CPU
REQUEST
READY
DATA / ADDRESS
I/O
INTERFACE
90 ,136
EXTERNAL MEMORY BUS

FIG._17

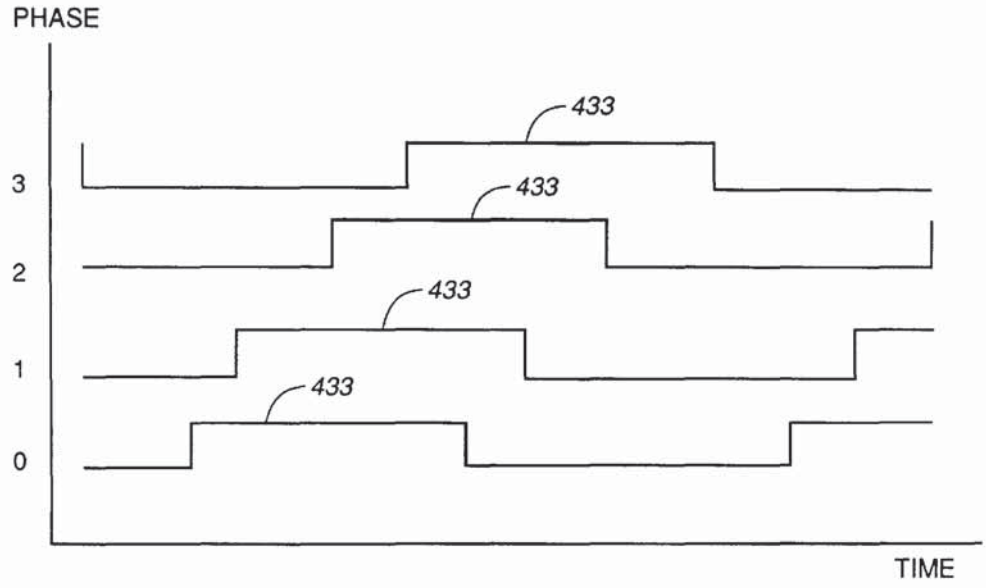

PHASE
433
433
433
433
3
2
1
0
TIME

FIG._19

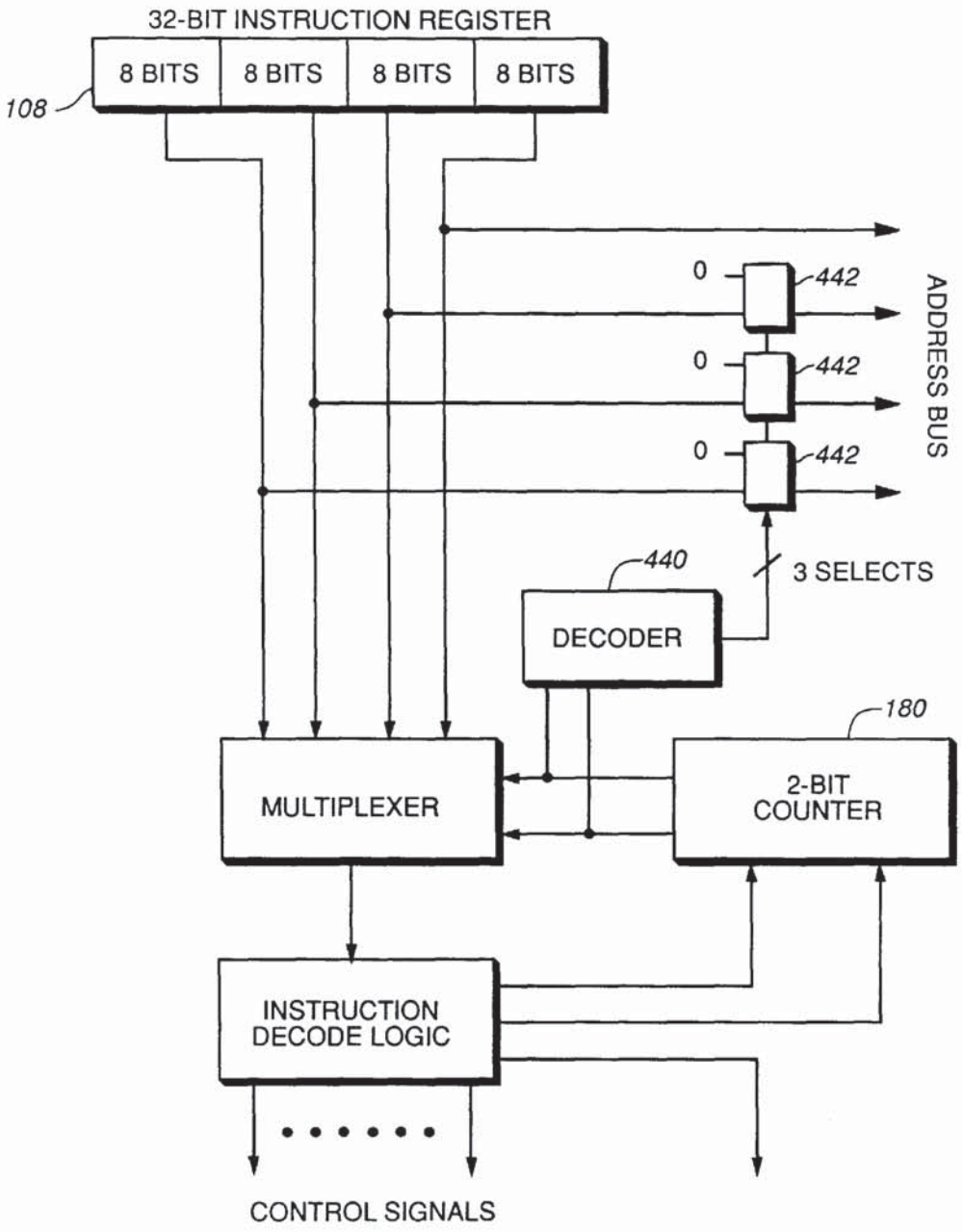

32-BIT INSTRUCTION REGISTER
8 BITS
8 BITS
8 BITS
8 BITS
108
0
442
0
442
0
442
ADDRESS BUS
440
3 SELECTS
DECODER
180
MULTIPLEXER
2-BIT COUNTER
INSTRUCTION DECODE LOGIC
CONTROL SIGNALS

FIG._20

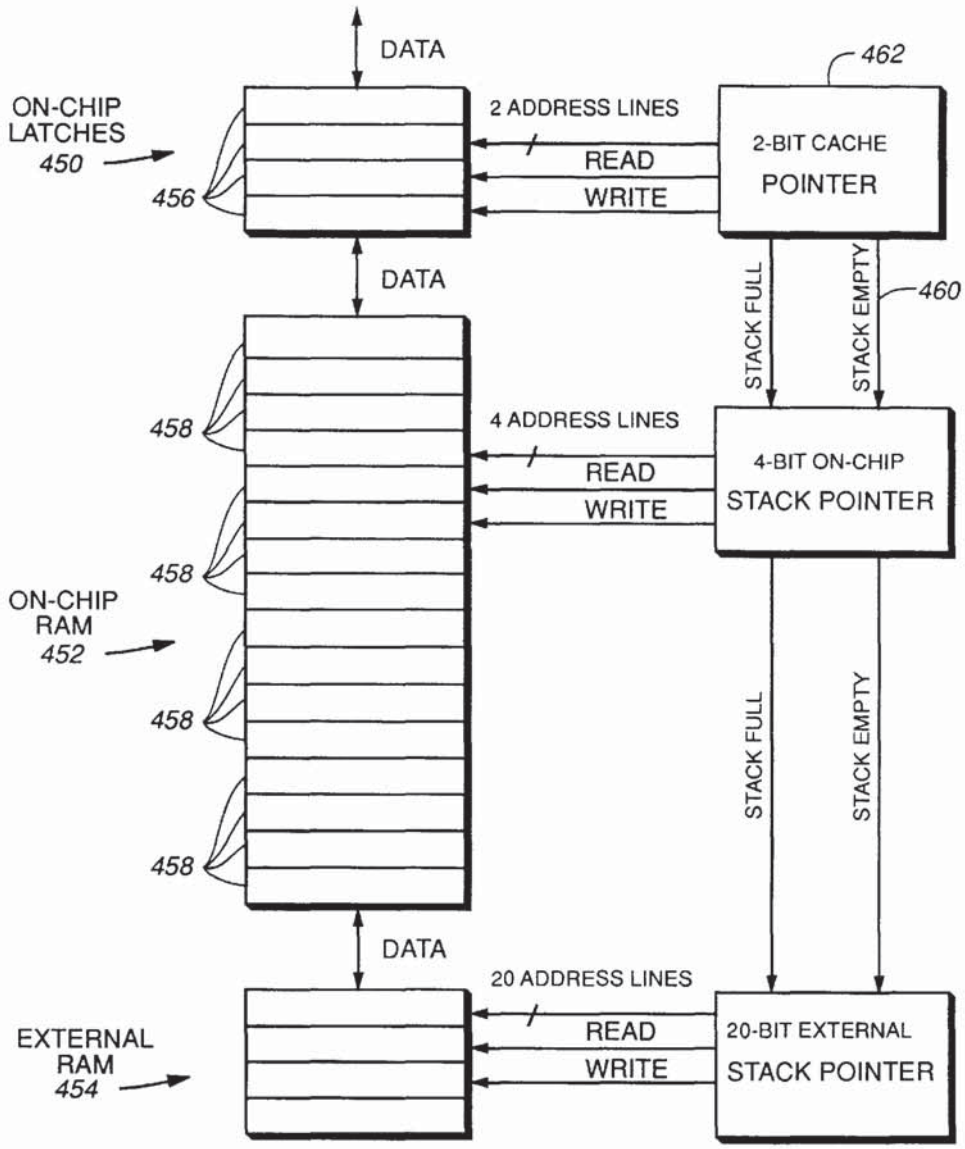

DATA
462
ON-CHIP LATCHES 450
2 ADDRESS LINES
READ
WRITE
2-BIT CACHE POINTER
456
DATA
STACK FULL
STACK EMPTY
460
458
4 ADDRESS LINES
READ
WRITE
4-BIT ON-CHIP STACK POINTER
458
ON-CHIP RAM 452
458
STACK FULL
STACK EMPTY
458
DATA
20 ADDRESS LINES
READ
WRITE
20-BIT EXTERNAL STACK POINTER
EXTERNAL RAM 454

FIG._21

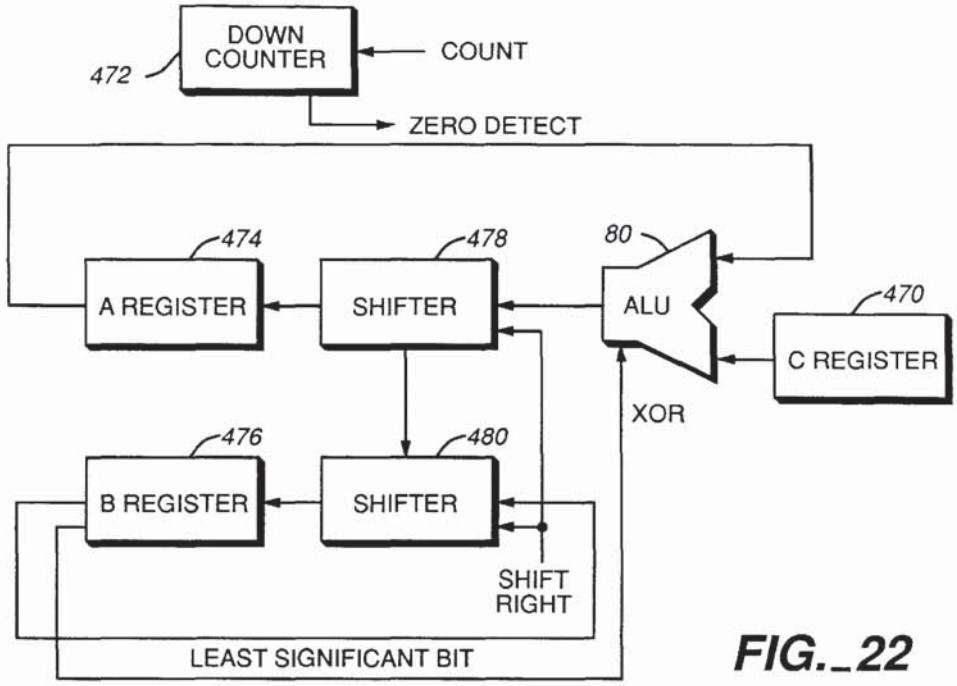

DOWN COUNTER
472
COUNT
ZERO DETECT
474
A REGISTER
478
SHIFTER
80
ALU
470
C REGISTER
XOR
476
B REGISTER
480
SHIFTER
SHIFT RIGHT
LEAST SIGNIFICANT BIT

FIG._22

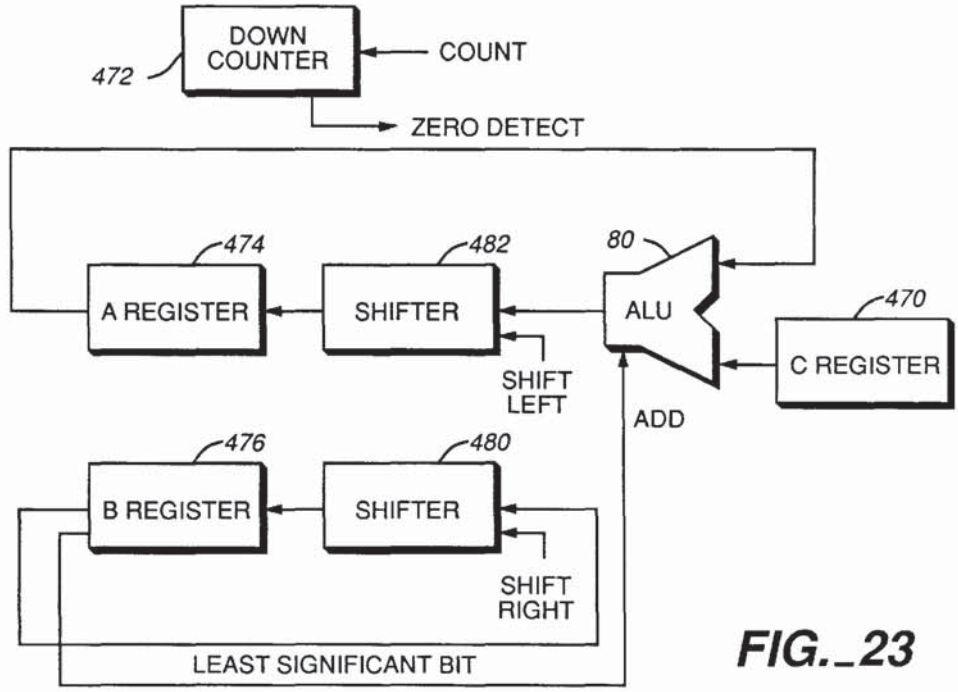

DOWN COUNTER
472
COUNT
ZERO DETECT
474
A REGISTER
482
SHIFTER
80
ALU
470
C REGISTER
SHIFT LEFT
ADD
476
B REGISTER
480
SHIFTER
SHIFT RIGHT
LEAST SIGNIFICANT BIT

FIG._23

HIGH PERFORMANCE MICROPROCESSOR HAVING VARIABLE SPEED SYSTEM CLOCK

CROSS REFERENCE TO RELATED APPLICATIONS

This application is a division of U.S. application Ser. No. 07/389,334, filed Aug. 3, 1989, now U.S. Pat. No. 5,440,749.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates generally to a simplified, reduced instruction set computer (RISC) microprocessor. More particularly, it relates to such a microprocessor which is capable of performance levels of, for example, 20 million instructions per second (MIPS) at a price of, for example, 20 dollars.

2. Description of the Prior Art

Since the invention of the microprocessor, improvements in its design have taken two different approaches. In the first approach, a brute force gain in performance has been achieved through the provision of greater numbers of faster transistors in the microprocessor integrated circuit and an instruction set of increased complexity. This approach is exemplified by the Motorola 68000 and Intel 80X86 microprocessor families. The trend in this approach is to larger die sizes and packages, with hundreds of pinouts.

More recently, it has been perceived that performance gains can be achieved through comparative simplicity, both in the microprocessor integrated circuit itself and in its instruction set. This second approach provides RISC microprocessors, and is exemplified by the Sun SPARC and the Intel 8960 microprocessors. However, even with this approach as conventionally practiced, the packages for the microprocessor are large, in order to accommodate the large number of pinouts that continue to be employed. A need therefore remains for further simplification of high performance microprocessors.

With conventional high performance microprocessors, fast static memories are required for direct connection to the microprocessors in order to allow memory accesses that are fast enough to keep up with the microprocessors. Slower dynamic random access memories (DRAMs) are used with such microprocessors only in a hierarchical memory arrangement, with the static memories acting as a buffer between the microprocessors and the DRAMs. The necessity to use static memories increases cost of the resulting systems.

Conventional microprocessors provide direct memory accesses (DMA) for system peripheral units through DMA controllers, which may be located on the microprocessor integrated circuit, or provided separately. Such DMA controllers can provide routine handling of DMA requests and responses, but some processing by the main central processing unit (CPU) of the microprocessor is required.

SUMMARY OF THE INVENTION

Accordingly, it is an object of this invention to provide a microprocessor with a reduced pin count and cost compared to conventional microprocessors.

It is another object of the invention to provide a high performance microprocessor that can be directly connected to DRAMs without sacrificing microprocessor speed.

It is a further object of the invention to provide a high performance microprocessor in which DMA does not require use of the main CPU during DMA requests and responses and which provides very rapid DMA response with predictable response times.

The attainment of these and related objects may be achieved through use of the novel high performance, low cost microprocessor herein disclosed. In accordance with one aspect of the invention, a microprocessor system in accordance with this invention has a central processing unit, a dynamic random access memory and a bus connecting the central processing unit to the dynamic random access memory. There is a multiplexing means on the bus between the central processing unit and the dynamic random access memory. The multiplexing means is connected and configured to provide row addresses, column addresses and data on the bus.

In accordance with another aspect of the invention, the microprocessor system has a means connected to the bus for fetching instructions for the central processing unit on the bus. The means for fetching instructions is configured to fetch multiple sequential instructions in a single memory cycle. In a variation of this aspect of the invention, a programmable read only memory containing instructions for the central processing unit is connected to the bus. The means for fetching instructions includes means for assembling a plurality of instructions from the programmable read only memory and storing the plurality of instructions in the dynamic random access memory.

In another aspect of the invention, the microprocessor system includes a central processing unit, a direct memory access processing unit and a memory connected by a bus. The direct memory access processing unit includes means for fetching instructions for the central processing unit and for fetching instructions for the direct memory access processing unit on the bus.

In a further aspect of the invention, the microprocessor system, including the memory, is contained in an integrated circuit. The memory is a dynamic random access memory, and the means for fetching multiple instructions includes a column latch for receiving the multiple instructions.

In still another aspect of the invention, the microprocessor system additionally includes an instruction register for the multiple instructions connected to the means for fetching instructions. A means is connected to the instruction register for supplying the multiple instructions in succession from the instruction register. A counter is connected to control the means for supplying the multiple instructions to supply the multiple instructions in succession. A means for decoding the multiple instructions is connected to receive the multiple instructions in succession from the means for supplying the multiple instructions. The counter is connected to said means for decoding to receive incrementing and reset control signals from the means for decoding. The means for decoding is configured to supply the reset control signal to the counter and to supply a control signal to the means for fetching instructions in response to a SKIP instruction in the multiple instructions. In a modification of this aspect of the invention, the microprocessor system additionally has a loop counter connected to receive a decrement control signal from the means for decoding. The means for decoding is configured to supply the reset control signal to the counter and the decrement control signal to the loop counter in response to a MICROLOOP instruction in the multiple instructions. In a further modification to this aspect of the invention, the means for decoding is configured to control

the counter in response to an instruction utilizing a variable width operand. A means is connected to the counter to select the variable width operand in response to the counter.

In a still further aspect of the invention, the microprocessor system includes an arithmetic logic unit. A first push down stack is connected to the arithmetic logic unit. The first push down stack includes means for storing a top item connected to a first input of the arithmetic logic unit and means for storing a next item connected to a second input of the arithmetic logic unit. The arithmetic logic unit has an output connected to the means for storing a top item. The means for storing a top item is connected to provide an input to a register file. The register file desirably is a second push down stack, and the means for storing a top item and the register file are bidirectionally connected.

In another aspect of the invention, a data processing system has a microprocessor including a sensing circuit and a driver circuit, a memory, and an output enable line connected between the memory, the sensing circuit and the driver circuit. The sensing circuit is configured to provide a ready signal when the output enable line reaches a predetermined electrical level, such as a voltage. The microprocessor is configured so that the driver circuit provides an enabling signal on the output enable line responsive to the ready signal.

In a further aspect of the invention, the microprocessor system has a ring counter variable speed system clock connected to the central processing unit. The central processing unit and the ring counter variable speed system clock are provided in a single integrated circuit. An input/output interface is connected to exchange coupling control signals, addresses and data with the input/output interface. A second clock independent of the ring counter variable speed system clock is connected to the input/output interface.

In yet another aspect of the invention, a push down stack is connected to the arithmetic logic unit. The push down stack includes means for storing a top item connected to a first input of the arithmetic logic unit and means for storing a next item connected to a second input of the arithmetic logic unit. The arithmetic logic unit has an output connected to the means for storing a top item. The push down stack has a first plurality of stack elements configured as latches and a second plurality of stack elements configured as a random access memory. The first and second plurality of stack elements and the central processing unit are provided in a single integrated circuit. A third plurality of stack elements is configured as a random access memory external to the single integrated circuit. In this aspect of the invention, desirably a first pointer is connected to the first plurality of stack elements, a second pointer connected to the second plurality of stack elements, and a third pointer is connected to the third plurality of stack elements. The central processing unit is connected to pop items from the first plurality of stack elements. The first stack pointer is connected to the second stack pointer to pop a first plurality of items from the second plurality of stack elements when the first plurality of stack elements are empty from successive pop operations by the central processing unit. The second stack pointer is connected to the third stack pointer to pop a second plurality of items from the third plurality of stack elements when the second plurality of stack elements are empty from successive pop operations by the central processing unit.

In another aspect of the invention, a first register is connected to supply a first input to the arithmetic logic unit. A first shifter is connected between an output of the arithmetic logic unit and the first register. A second register is connected to receive a starting polynomial value. An output of the second register is connected to a second shifter. A least significant bit of the second register is connected to The arithmetic logic unit. A third register is connected to supply feedback terms of a polynomial to the arithmetic logic unit. A down counter, for counting down a number corresponding to digits of a polynomial to be generated, is connected to the arithmetic logic unit. The arithmetic logic unit is responsive to a polynomial instruction to carry out an exclusive OR of the contents of the first register with the contents of the third register if the least significant bit of the second register is a "ONE" and to pass the contents of the first register unaltered if the least significant bit of the second register is a "ZERO", until the down counter completes a count. The polynomial to be generated results in said first register.

In still another aspect of the invention, a result register is connected to supply a first input to the arithmetic logic unit. A first, left shifting shifter is connected between an output of the arithmetic logic unit and the result register. A multiplier register is connected to receive a multiplier in bit reversed form. An output of the multiplier register is connected to a second, right shifting shifter. A least significant bit of the multiplier register is connected to the arithmetic logic unit. A third register is connected to supply a multiplicand to said arithmetic logic unit. A down counter, for counting down a number corresponding to one less than the number of digits of the multiplier, is connected to the arithmetic logic unit. The arithmetic logic unit is responsive to a multiply instruction to add the contents of the result register with the contents of the third register, when the least significant bit of the multiplier register is a "ONE" and to pass the contents of the result register unaltered, until the down counter completes a count. The product results in the result register.

The attainment of the foregoing and related objects, advantages and features of the invention should be more readily apparent to those skilled in the art, after review of the following more detailed description of the invention, taken together with the drawings, in which:

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. **1** is an external, plan view of an integrated circuit package incorporating a microprocessor in accordance with the invention.

FIG. **2** is a block diagram of a microprocessor in accordance with the invention.

FIG. **3** is a block diagram of a portion of a data processing system incorporating the microprocessor of FIGS. **1** and **2**.

FIG. **4** is a more detailed block diagram of a portion of the microprocessor shown in FIG. **2**.

FIG. **5** is a more detailed block diagram of another portion of the microprocessor shown in FIG. **2**.

FIG. **6** is a block diagram of another portion of the data processing system shown in part in FIG. **3** and incorporating the microprocessor of FIGS. **1–2** and **4–5**.

FIGS. **7** and **8** are layout diagrams for the data processing system shown in part in FIGS. **3** and **6**.

FIG. **9** is a layout diagram of a second embodiment of a microprocessor in accordance with the invention in a data processing system on a single integrated circuit.

FIG. **10** is a more detailed block diagram of a portion of the data processing system of FIGS. **7** and **8**.

FIG. **11** is a timing diagram useful for understanding operation of the system portion shown in FIG. **12**.

FIG. **12** is another more detailed block diagram of a further portion of the data processing system of FIGS. **7** and **8**.

FIG. **13** is a more detailed block diagram of a portion of the microprocessor shown in FIG. **2**.

FIG. **14** is a more detailed block and schematic diagram of a portion of the system shown in FIGS. **3** and **7–8**.

FIG. **15** is a graph useful for understanding operation of the system portion shown in FIG. **14**.

FIG. **16** is a more detailed block diagram showing part of the system portion shown in FIG. **4**.

FIG. **17** is a more detailed block diagram of a portion of the microprocessor shown in FIG. **2**.

FIG. **18** is a more detailed block diagram of part of the microprocessor portion shown in FIG. **17**.

FIG. **19** is a set of waveform diagrams useful for understanding operation of the part of the microprocessor portion shown in FIG. **18**.

FIG. **20** is a more detailed block diagram showing another part of the system portion shown in FIG. **4**.

FIG. **21** is a more detailed block diagram showing another part of the system portion shown in FIG. **4**.

FIGS. **22** and **23** are more detailed block diagrams showing another part of the system portion shown in FIG. **4**.

DETAILED DESCRIPTION OF THE INVENTION

Overveiw

The microprocessor of this invention is desirably implemented as a 32-bit microprocessor optimized for:

HIGH EXECUTION SPEED, and

LOW SYSTEM COST.

In this embodiment, the microprocessor can be thought of as 20 MIPS for 20 dollars. Important distinguishing features of the microprocessor are:

Uses low-cost commodity DYNAMIC RAMS to run 20 MIPS

4 instruction fetch per memory cycle

On-chip fast page-mode memory management

Runs fast without external cache

Requires few interfacing chips

Crams 32-bit CPU in 44 pin SOJ package

The instruction set is organized so that most operations can be specified with 8-bit instructions. Two positive products of this philosophy are:

Programs are smaller,

Programs can execute much faster.

The bottleneck in most computer systems is the memory bus. The bus is used to fetch instructions and fetch and store data. The ability to fetch four instructions in a single memory bus cycle significantly increases the bus availability to handle data.

Turning now to the drawings, more particularly to FIG. **1**, there is shown a packaged 32-bit microprocessor **50** in a 44-pin plastic leadless chip carrier, shown approximately 100 times its actual size of about 0.8 inch on a side. The fact that the microprocessor **50** is provided as a 44-pin package represents a substantial departure from typical microprocessor packages, which usually have about 200 input/output (I/O) pins. The microprocessor **50** is rated at 20 million instructions per second (MIPS). Address and data lines **52**, also labelled D**0**–D**31**, are shared for addresses and data without speed penalty as a result of the manner in which the microprocessor **50** operates, as will be explained below.

DYNAMIC RAM

In addition to the low cost 44-pin package, another unusual aspect of the high performance microprocessor **50** is that it operates directly with dynamic random access memories (DRAMs), as shown by row address strobe (RAS) and column address strobe (CAS) I/O pins **54**. The other I/O pins for the microprocessor **50** include V_{DD} pins **56**, V_{SS} pins **58**, output enable pin **60**, write pin **62**, clock pin **64** and reset pin **66**.

All high speed computers require high speed and expensive memory to keep up. The highest speed static RAM memories cost as much as ten times as much as slower dynamic RAMs. This microprocessor has been optimized to use low-cost dynamic RAM in high-speed page-mode. Page-mode dynamic RAMs offer static RAM performance without the cost penalty. For example, low-cost 85 nsec. dynamic RAMs access at 25 nsec when operated in fast page-mode. Integrated fast page-mode control on the microprocessor chip simplifies system interfacing and results in a faster system.

Details of the microprocessor **50** are shown in FIG. **2**. The microprocessor **50** includes a main central processing unit (CPU) **70** and a separate direct memory access (DMA) CPU **72** in a single integrated circuit making up the microprocessor **50**. The main CPU **70** has a first **16** deep push down stack **74**, which has a top item register **76** and a next item register **78**, respectively connected to provide inputs to an arithmetic logic unit (ALU) **80** by lines **82** and **84**. An output of the ALU **80** is connected to the top item register **76** by line **86**. The output of the top item register at **82** is also connected by line **88** to an internal data bus **90**.

A loop counter **92** is connected to a decrementer **94** by lines **96** and **98**. The loop counter **92** is bidirectionally connected to the internal data bus **90** by line **100**. Stack pointer **102**, return stack pointer **104**, mode register **106** and instruction register **108** are also connected to the internal data bus **90** by lines **110**, **112**, **114** and **116**, respectively. The internal data bus **90** is connected to memory controller **118** and to gate **120**. The gate **120** provides inputs on lines **122**, **124**, and **126** to X register **128**, program counter **130** and Y register **132** of return push down stack **134**. The X register **128**, program counter **130** and Y register **132** provide outputs to internal address bus **136** on lines **138**, **140** and **142**. The internal address bus provides inputs to the memory controller **118** and to an incrementer **144**. The incrementer **144** provides inputs to the X register, program counter and Y register via lines **146**, **122**, **124** and **126**. The DMA CPU **72** provides inputs to the memory controller **118** on line **148**. The memory controller **118** is connected to a RAM (not shown) by address/data bus **150** and control lines **152**.

FIG. **2** shows that the microprocessor **50** has a simple architecture. Prior art RISC microprocessors are substantially more complex in design. For example, the SPARC RISC microprocessor has three times the gates of the microprocessor **50**, and the Intel 8960 RISC microprocessor has 20 times the gates of the microprocessor **50**. The speed of this microprocessor is in substantial part due to this simplicity. The architecture incorporates push down stacks and register write to achieve this simplicity.

The microprocessor **50** incorporates an I/O that has been tuned to make heavy use of resources provided on the integrated circuit chip. On chip latches allow use of the same I/O circuits to handle three different things: column addressing, row addressing and data, with a slight to non-existent speed penalty. This triple bus multiplexing results in fewer buffers to expand, fewer interconnection lines, fewer I/O pins and fewer internal buffers.

The provision of on-chip DRAM control gives a performance equal to that obtained with the use of static RAMs. As a result, memory is provided at ¼ the system cost of static RAM used in most RISC systems.

The microprocessor **50** fetches 4 instructions per memory cycle; the instructions are in an 8-bit format, and this is a 32-bit microprocessor. System speed is therefore 4 times the memory bus bandwidth. This ability enables the microprocessor to break the Von Neumann bottleneck of the speed of getting the next instruction. This mode of operation is possible because of the use of a push down stack and register array. The push down stack allows the use of implied addresses, rather than the prior art technique of explicit addresses for two sources and a destination.

Most instructions execute in 20 nanoseconds in the microprocessor **50**. The microprocessor can therefore execute instructions at 50 peak MIPS without pipeline delays. This is a function of the small number of gates in the microprocessor **50** and the high degree of parallelism in the architecture of the microprocessor.

FIG. **3** shows how column and row addresses are multiplexed on lines D**8**–D**14** of the microprocessor **50** for addressing DRAM **150** from I/O pins **52**. The DRAM **150** is one of eight, but only one DRAM **150** has been shown for clarity. As shown, the lines D**11**–D**18** are respectively connected to row address inputs A**0**–A**8** of the DRAM **150**. Additionally, lines D**12**–D**15** are connected to the data inputs DQ**1**–DQ**4** of the DRAM **150**. The output enable, write and column address strobe pins **54** are respectively connected to the output enable, write and column address strobe inputs of the DRAM **150** by lines **152**. The row address strobe pin **54** is connected through row address strobe decode logic **154** to the row address strobe input of the DRAM **150** by lines **156** and **158**.

D**0**–D**7** pins **52** (FIG. **1**) are idle when the microprocessor **50** is outputting multiplexed row and column addresses on D**11**–D**18** pins **52**. The D**0**–D**7** pins **52** can therefore simultaneously be used for I/O when right justified I/O is desired. Simultaneous addressing and I/O can therefore be carried out.

FIG. **4** shows how the microprocessor **50** is able to achieve performance equal to the use of static RAMS with DRAMs through multiple instruction fetch in a single clock cycle and instruction fetch-ahead. Instruction register **108** receives four 8-bit byte instruction words 1–4 on 32-bit internal data bus **90**. The four instruction byte 1–4 locations of the instruction register **108** are connected to multiplexer **170** by busses **172**, **174**, **176** and **178**, respectively. A microprogram counter **180** is connected to the multiplexer **170** by lines **182**. The multiplexer **170** is connected to decoder **184** by bus **186**. The decoder **184** provides internal signals to the rest of the microprocessor **50** on lines **188**.

Most significant bits **190** of each instruction byte 1–4 location are connected to a 4-input decoder **192** by lines **194**. The output of decoder **192** is connected to memory controller **118** by line **196**. Program counter **130** is connected to memory controller **118** by internal address bus **136**, and the instruction register **108** is connected to the memory controller **118** by the internal data bus **90**. Address/data bus **198** and control bus **200** are connected to the DRAMS **150** (FIG. **3**).

In operation, when the most significant bits **190** of remaining instructions 1–4 are "1" in a clock cycle of the microprocessor **50**, there are no memory reference instructions in the queue. The output of decoder **192** on line **196** requests an instruction fetch ahead by memory controller **118** without interference with other accesses. While the current instructions in instruction register **108** are executing, the memory controller **118** obtains the address of the next set of four instructions from program counter **130** and obtains that set of instructions. By the time the current set of instructions has completed execution, the next set of instructions is ready for loading into the instruction register.

Details of the DMA CPU **72** are provided in FIG. **5**. Internal data bus **90** is connected to memory controller **118** and to DMA instruction register **210**. The DMA instruction register **210** is connected to DMA program counter **212** by bus **214**, to transfer size counter **216** by bus **218** and to timed transfer interval counter **220** by bus **222**. The DMA instruction register **210** is also connected to DMA I/O and RAM address register **224** by line **226**. The DMA I/O and RAM address register **224** is connected to the memory controller **118** by memory cycle request line **228** and bus **230**. The DMA program counter **212** is connected to the internal address bus **136** by bus **232**. The transfer size counter **216** is connected to a DMA instruction done decrementer **234** by lines **236** and **238**. The decrementer **234** receives a control input on memory cycle acknowledge line **240**. When transfer size counter **216** has completed its count, it provides a control signal to DMA program counter **212** on line **242**. Timed transfer interval counter **220** is connected to decrementer **244** by lines **246** and **248**. The decrementer **244** receives a control input from a microprocessor system clock on line **250**.

The DMA CPU **72** controls itself and has the ability to fetch and execute instructions. It operates as a co-processor to the main CPU **70** (FIG. **2**) for time specific processing.

FIG. **6** shows how the microprocessor **50** is connected to an electrically programmable read only memory (EPROM) **260** by reconfiguring the data lines **52** so that some of the data lines **52** are input lines and some of them are output lines. Data lines **52** D**0**–D**7** provide data to and from corresponding data terminals **262** of the EPROM **260**. Data lines **52** D**9**–D**18** provide addresses to address terminals **264** of the EPROM **260**. Data lines **52** D**19**–D**31** provide inputs from the microprocessor **50** to memory and I/O decode logic **266**. RAS 0/1 control line **268** provides a control signal for determining whether the memory and I/O decode logic provides a DRAM RAS output on line **270** or a column enable output for the EPROM **260** on line **272**. Column address strobe terminal **60** of the microprocessor **50** provides an output enable signal on line **274** to the corresponding terminal **276** of the EPROM **260**.

FIGS. **7** and **8** show the front and back of a one card data processing system **280** incorporating the microprocessor **50**, MSM514258-10 type DRAMs **150** totalling 2 megabytes, a Motorola 50 MegaHertz crystal oscillator clock **282**, I/O circuits **284** and a 27256 type EPROM **260**. The I/O circuits **284** include a 74HC04 type high speed hex inverter circuit **286**, an IDT39C828 type 10-bit inverting buffer circuit **288**, an IDT39C822 type 10-bit inverting register circuit **290**, and two IDT39C823 type 9-bit non-inverting register circuits **292**. The card **280** is completed with a MAX12V type DC-DC converter circuit **294**, 34-pin dual AMP type headers **296**, a coaxial female power connector **298**, and a 3-pin AMP right angle header **300**. The card **280** is a low cost, imbeddable product that can be incorporated in larger systems or used as an internal development tool.

The microprocessor **50** is a very high performance (50 MHz) RISC influenced 32-bit CPU designed to work closely with dynamic RAM. Clock for clock, the microprocessor **50** approaches the theoretical performance limits possible with a single CPU configuration. Eventually, the microprocessor **50** and any other processor is limited by the bus bandwidth and the number of bus paths. The critical conduit is between the CPU and memory.

One solution to the bus bandwidth/bus path problem is to integrate a CPU directly onto the memory chips, giving every memory a direct bus the CPU. FIG. **9** shows another microprocessor **310** that is provided integrally with 1 mega-

bit of DRAM **311** in a single integrated circuit **312**. Until the present invention, this solution has not been practical, because most high performance CPUs require from 500,000 to 1,000,000 transistors and enormous die sizes just by themselves. The microprocessor **310** is equivalent to the microprocessor **50** in FIGS. **1–8**. The microprocessors **50** and **310** are the most transistor efficient high performance CPUs in existence, requiring fewer than 50,000 transistors for dual processors **70** and **72** (FIG. **2**) or **314** and **316** (less memory). The very high speed of the microprocessors **50** and **310** is to a certain extent a function of the small number of active devices. In essence, the less silicon gets in the way, the faster the electrons can get where they are going.

The microprocessor **310** is therefore the only CPU suitable for integration on the memory chip die **312**. Some simple modifications to the basic microprocessor **50** to take advantage of the proximity to the DRAM array **311** can also increase the microprocessor **50** clock speed by **50** percent, and probably more.

The microprocessor **310** core on board the DRAM die **312** provides most of the speed and functionality required for a large group of applications from automotive to peripheral control. However, the integrated CPU **310**/DRAM **311** concept has the potential to redefine significantly the way multiprocessor solutions can solve a spectrum of very compute intensive problems. The CPU **310**/DRAM **311** combination eliminates the Von Neumann bottleneck by distributing it across numerous CPU/DRAM chips **312**. The microprocessor **310** is a particularly good core for multiprocessing, since it was designed with the SDI targeting array in mind, and provisions were made for efficient interprocessor communications.

Traditional multiprocessor implementations have been very expensive in addition to being unable to exploit fully the available CPU horsepower. Multiprocessor systems have typically been built up from numerous board level or box level computers. The result is usually an immense amount of hardware with corresponding wiring, power consumption and communications problems. By the time the systems are interconnected, as much as 50 percent of the bus speed has been utilized just getting through the interfaces.

In addition, multiprocessor system software has been scarce. A multiprocessor system can easily be crippled by an inadequate load-sharing algorithm in the system software, which allows one CPU to do a great deal of work and the others to be idle. Great strides have been made recently in systems software, and even UNIX V.4 may be enhanced to support multiprocessing. Several commercial products from such manufacturers as DUAL Systems and UNISOFT do a credible job on 68030 type microprocessor systems now.

The microprocessor **310** architecture eliminates most of the interface friction, since up to 64 CPU **310**/RAM **311** processors should be able to intercommunicate without buffers or latches. Each chip **312** has about 40 MIPS raw speed, because placing the DRAM **311** next to the CPU **310** allows the microprocessor **310** instruction cycle to be cut in half, compared to the microprocessor **50**. A 64 chip array of these chips **312** is more powerful than any other existing computer. Such an array fits on a 3×5 card, cost less than a FAX machine, and draw about the same power as a small television.

Dramatic changes in price/performance always reshape existing applications and almost always create new ones. The introduction of microprocessors in the mid 1970s created video games, personal computers, automotive computers, electronically controlled appliances, and low cost computer peripherals.

The integrated circuit **312** will find applications in all of the above areas, plus create some new ones. A common generic parallel processing algorithm handles convolution/Fast Fourier Transform (FFT)/pattern recognition. Interesting product possibilities using the integrated circuit **312** include high speed reading machines, real-time speech recognition, spoken language translation, real-time robot vision, a product to identify people by their faces, and an automotive or aviation collision avoidance system.

A real time processor for enhancing high density television (HDTV) images, or compressing the HDTV information into a smaller bandwidth, would be very. feasible. The load sharing in HDTV could be very straightforward. Splitting up the task according to color and frame would require 6, 9 or 12 processors. Practical implementation might require 4 meg RAMs integrated with the microprocessor **310**.

The microprocessor **310** has the following specifications:

CONTROL LINES
4—POWER/GROUND
1—CLOCK
32—DATA I/O
4—SYSTEM CONTROL
EXTERNAL MEMORY FETCH
EXTERNAL MEMORY FETCH AUTOINCREMENT X
EXTERNAL MEMORY FETCH AUTOINCREMENT Y
EXTERNAL MEMORY WRITE
EXTERNAL MEMORY WRITE AUTOINCREMENT X
EXTERNAL MEMORY WRITE AUTOINCREMENT Y
EXTERNAL PROM FETCH
LOAD ALL X REGISTERS
LOAD ALL Y REGISTERS
LOAD ALL PC REGISTERS
EXCHANGE X AND Y
INSTRUCTION FETCH
ADD TO PC
ADD TO X
WRITE MAPPING REGISTER
READ MAPPING REGISTER
REGISTER CONFIGURATION
MICROPROCESSOR **310** CPU **316** CORE
COLUMN LATCH1 (1024 BITS) 32×32 MUX
STACK POINTER (16 BITS)
COLUMN LATCH2 (1024 BITS) 32×32 MUX
RSTACK POINTER (16 BITS)
PROGRAM COUNTER 32 BITS
X0 REGISTER 32 BITS (ACTIVATED ONLY FOR ON-CHIP ACCESSES)
Y0 REGISTER 32 BITS (ACTIVATED ONLY FOR ON-CHIP ACCESSES)
LOOP COUNTER 32 BITS
DMA CPU **314** CORE
DMA PROGRAM COUNTER 24 BITS
INSTRUCTION REGISTER 32 BITS
I/O & RAM ADDRESS REGISTER 32 BITS
TRANSFER SIZE COUNTER 12 BITS
INTERVAL COUNTER 12 BITS

To offer memory expansion for the basic chip **312**, an intelligent DRAM can be produced. This chip will be optimized for high speed operation with the integrated circuit **312** by having three on-chip address registers: Program Counter, X Register and Y register. As a result, to access the intelligent DRAM, no address is required, and a total access cycle could be as short as 10 nsec. Each

expansion DRAM would maintain its own copy of the three registers and would be identified by a code specifying its memory address. Incrementing and adding to the three registers will actually take place on the memory chips. A maximum of 64 intelligent DRAM peripherals would allow a large system to be created without sacrificing speed by introducing multiplexers or buffers.

There are certain differences between the microprocessor **310** and the microprocessor **50** that arise from providing the microprocessor **310** on the same die **312** with the DRAM **311**. Integrating the DRAM **311** allows architectural changes in the microprocessor **310** logic to take advantage of existing on-chip DRAM **311** circuitry. Row and column design is inherent in memory architecture. The DRAMs **311** access random bits in a memory array by first selecting a row of 1024 bits, storing them into a column latch, and then selecting one of the bits as the data to be read or written.

The time required to access the data is split between the row access and the column access. Selecting data already stored in a column latch is faster than selecting a random bit by at least a factor of six. The microprocessor **310** takes advantage of this high speed by creating a number of column latches and using them as caches and shift registers. Selecting a new row of information may be thought of as performing a 1024-bit read or write with the resulting immense bus bandwidth.

1. The microprocessor **50** treats its 32-bit instruction register **108** (see FIGS. **2** and **4**) as a cache for four 8-bit instructions. Since the DRAM **311** maintains a 1024-bit latch for the column bits, the microprocessor **310** treats the column latch as a cache for 128 8-bit instructions. Therefore, the next instruction will almost always be already present in the cache. Long loops within the cache are also possible and more useful than the 4 instruction loops in the microprocessor **50**.

2. The microprocessor **50** uses two 16×32-bit deep register arrays **74** and **134** (FIG. **2**) for the parameter stack and the return stack. The microprocessor **310** creates two other 1024-bit column latches to provide the equivalent of two 32×32-bit arrays, which can be accessed twice as fast as a register array.

3. The microprocessor **50** has a DMA capability which can be used for I/O to a video shift register. The microprocessor **310** uses yet another 1024-bit column latch as a long video shift register to drive a CRT display directly. For color displays, three on-chip shift registers could also be used. These shift registers can transfer pixels at a maximum of 100 MHz.

4. The microprocessor **50** accesses memory via an external 32-bit bus. Most of the memory **311** for the microprocessor **310** is on the same die **312**. External access to more memory is made using an 8-bit bus. The result is a smaller die, smaller package and lower power consumption than the microprocessor **50**.

5. The microprocessor **50** consumes about a third of its operating power charging and discharging the I/O pins and associated capacitances. The DRAMs **150** (FIG. **8**) connected to the microprocessor **50** dissipate most of their power in the I/O drivers. A microprocessor **310** system will consume about one-tenth the power of a microprocessor **50** system, since having the DRAM **311** next to the processor **310** eliminates most of the external capacitances to be charged and discharged.

6. Multiprocessing means splitting a computing task between numerous processors in order to speed up the solution. The popularity of multiprocessing is limited by the expense of current individual processors as well as the limited interprocessor communications ability. The microprocessor **310** is an excellent multiprocessor candidate, since the chip **312** is a monolithic computer complete with memory, rendering it low-cost and physically compact.

The shift registers implemented with the microprocessor **310** to perform video output can also be configured as interprocessor communication links. The INMOS transputer attempted a similar strategy, but at much lower speed and without the performance benefits inherent in the microprocessor **310** column latch architecture. Serial I/O is a prerequisite for many multiprocessor topologies because of the many neighbor processors which communicate. A cube has 6 neighbors. Each neighbor communicates using these lines:

DATA IN
CLOCK IN
READY FOR DATA
DATA OUT
DATA READY?
CLOCK OUT

A special start up sequence is used to initialize the on-chip DRAM **311** in each of the processors.

The microprocessor **310** column latch architecture allows neighbor processors to deliver information directly to internal registers or even instruction caches of other chips **312**. This technique is not used with existing processors, because it only improves performance in a tightly coupled DRAM system.

7. The microprocessor **50** architecture offers two types of looping structures: LOOP-IF-DONE and MICRO-LOOP. The former takes an 8-bit to 24-bit operand to describe the entry point to the loop address. The latter performs a loop entirely within the 4 instruction queue and the loop entry point is implied as the first instruction in the queue. Loops entirely within the queue run without external instruction fetches and execute up to three times as fast as the long loop construct. The microprocessor **310** retains both constructs with a few differences. The microprocessor **310** microloop functions in the same fashion as the microprocessor **50** operation, except the queue is 1024-bits or 128 8-bit instructions long. The microprocessor **310** microloop can therefore contain jumps, branches, calls and immediate operations not possible in the 4 8-bit instruction microprocessor **50** queue.

Microloops in the microprocessor **50** can only perform simple block move and compare functions. The larger microprocessor **310** queue allows entire digital signal processing or floating point algorithms to loop at high speed in the queue.

The microprocessor **50** offers four instructions to redirect execution:

CALL
BRANCH
BRANCH-IF-ZERO
LOOP-IF-NOT-DONE

These instructions take a variable length address operand 8, 16 or 24 bits long. The microprocessor **50** next address logic treats the three operands similarly by adding or subtracting them to the current program counter. For the microprocessor **310**, the 16 and 24-bit operands function in the same manner as the 16 and 24-bit operands in the microprocessor **50**. The 8-bit class operands are reserved to operate entirely within the instruction queue. Next address decisions can therefore be made quickly, because only 10 bits of addresses are affected, rather than 32. There is no carry or borrow generated past the 10 bits.

8. The microprocessor **310** CPU **316** resides on an already crowded DRAM die **312**. To keep chip size as small as

possible, the DMA processor **72** of the microprocessor **50** has been replaced with a more traditional DMA controller **314**. DMA is used with the microprocessor **310** to perform the following functions:

Video output to a CRT

Multiprocessor serial communications

8-bit parallel I/O

The DMA controller **314** can maintain both serial and parallel transfers simultaneously. The following DMA sources and destinations are supported by the microprocessor **310**:

DESCRIPTION	I/O	LINES
1. Video shift register	OUTPUT	1 to 3
2. Multiprocessor serial	BOTH	6 lines/channel
3. 8-bit parallel	BOTH	8 data, 4 control

The three sources use separate 1024-bit buffers and separate I/O pins. Therefore, all three may be active simultaneously without interference.

The microprocessor **310** can be implemented with either a single multiprocessor serial buffer or separate receive and sending buffers for each channel, allowing simultaneous bidirectional communications with six neighbors simultaneously.

FIGS. **10** and **11** provide details of the PROM DMA used in the microprocessor **50**. The microprocessor **50** executes faster than all but the fastest PROMs. PROMS are used in a microprocessor **50** system to store program segments and perhaps entire programs. The microprocessor **50** provides a feature on power-up to allow programs to be loaded from low-cost, slow speed PROMs into high speed DRAM for execution. The logic which performs this function is part of the DMA memory controller **118**. The operation is similar to DMA, but not identical, since four 8-bit bytes must be assembled on the microprocessor **50** chip, then written to the DRAM **150**.

The microprocessor **50** directly interfaces to DRAM **150** over a triple multiplexed data and address bus **350**, which carries RAS addresses, CAS addresses and data. The EPROM **260**, on the other hand, is read with non-multiplexed busses. The microprocessor **50** therefore has a special mode which unmultiplexes the data and address lines to read 8 bits of EPROM data. Four 8-bit bytes are read in this fashion. The multiplexed bus **350** is turned back on, and the data is written to the DRAM **150**.

When the microprocessor **50** detects a RESET condition, the processor stops the main CPU **70** and forces a mode **0** (PROM LOAD) instruction into the DMA CPU **72** instruction register. The DMA instruction directs the memory controller to read the EPROM **260** data at 8 times the normal access time for memory. Assuming a 50 MHz microprocessor **50**, this means an access time of 320 nsec. The instruction also indicates:

The selection address of the EPROM **260** to be loaded,

The number of 32-bit words to transfer,

The DRAM **150** address to transfer into.

The sequence of activities to transfer one 32-bit word from EPROM **260** to DRAM **150** are:

1. RAS goes low at **352**, latching the EPROM **260** select information from the high order address bits. The EPROM **260** is selected.
2. Twelve address bits (consisting of what is normally DRAM CAS addresses plus two byte select bits are placed on the bus **350** going to the EPROM **260** address pins. These signals will remain on the lines until the data from the EPROM **260** has been read into the microprocessor **50**. For the first byte, the byte select bits will be binary 00.
3. CAS goes low at **354**, enabling the EPROM **260** data onto the lower 8 bits of the external address/data bus **350**. NOTE: It is important to recognize that, during this part of the cycle, the lower 8 bits of the external data/address bus are functioning as inputs, but the rest of the bus is still acting as outputs.
4. The microprocessor **50** latches these eight least significant bits internally and shifts them 8 bits left to shift them to the next significant byte position.
5. Steps 2, 3 and 4 are repeated with byte address 01.
6. Steps 2, 3 and 4 are repeated with byte address 10.
7. Steps 2, 3 and 4 are repeated with byte address 11.
8. CAS goes high at **356**, taking the EPROM **260** off the data bus.
9. RAS goes high at **358**, indicating the end of the EPROM **260** access.
10. RAS goes low at **360**, latching the DRAM select information from the high order address bits. At the same time, the RAS address bits are latched into the DRAM **150**. The DRAM **150** is selected.
11. CAS goes low at **362**, latching the DRAM **150** CAS addresses.
12. The microprocessor **50** places the previously latched EPROM **260** 32-bit data onto the external address/data bus **350**. W goes low at **364**, writing the 32 bits into the DRAM **150**.
13. W goes high at **366**. CAS goes high at **368**. The process continues with the next word.

FIG. **12** shows details of the microprocessor **50** memory controller **118**. In operation, bus requests stay present until they are serviced. CPU **70** requests are prioritized at **370** in the order of: 1, Parameter Stack; 2, Return Stack; 3, Data Fetch; 4, Instruction Fetch. The resulting CPU request signal and a DMA request signal are supplied as bus requests to bus control **372**, which provides a bus grant signal at **374**. Internal address bus **136** and a DMA counter **376** provide inputs to a multiplexer **378**. Either a row address or a column address are provided as an output to multiplexed address bus **380** as an output from the multiplexer **378**. The multiplexed address bus **380** and the internal data bus **90** provide address and data inputs, respectively, to multiplexer **382**. Shift register **384** supplies row address strobe (RAS) **1** and **2** control signals to multiplexer **386** and column address strobe (CAS) **1** and **2** control signals to multiplexer **388** on lines **390** and **392**. The shift register **384** also supplies output enable (OE) and write (W) signals on lines **394** and **396** and a control signal on line **398** to multiplexer **382**. The shift register **384** receives a RUN signal on line **400** to generate a memory cycle and supplies a MEMORY READY signal on line **402** when an access is complete.

STACK/REGISTER ARCHITECTURE

Most microprocessors use on-chip registers for temporary storage of variables. The on-chip registers access data faster than off-chip RAM. A few microprocessors use an on-chip push down stack for temporary storage.

A stack has the advantage of faster operation compared to on-chip registers by avoiding the necessity to select source and destination registers. (A math or logic operation always uses the top two stack items as source and the top of stack as destination.) The stack's disadvantage is that it makes some operations clumsy. Some compiler activities in particular require on-chip registers for efficiency.

As shown in FIG. **13**, the microprocessor **50** provides both on-chip registers **134** and a stack **74** and reaps the benefits of both.

BENEFITS:

1. Stack math and logic is twice as fast as those available on an equivalent register only machine. Most programmers and optimizing compilers can take advantage of this feature.
2. Sixteen registers are available for on-chip storage of local variables which can transfer to the stack for computation. The accessing of variables is three to four times as fast as available on a strictly stack machine.

The combined stack **74**/register **134** architecture has not been used previously due to inadequate understanding by computer designers of optimizing compilers and the mix of transfer versus math/logic instructions.

ADAPTIVE MEMORY CONTROLLER

A microprocessor must be designed to work with small or large memory configurations. As more memory loads are added to the data, address, and control lines, the switching speed of the signals slows down. The microprocessor **50** multiplexes the address/data bus three ways, so timing between the phases is critical. A traditional approach to the problem allocates a wide margin of time between bus phases so that systems will work with small or large numbers of memory chips connected. A speed compromise of as much as 50% is required.

As shown in FIG. **14**, the microprocessor **50** uses a feedback technique to allow the processor to adjust memory bus timing to be fast with small loads and slower with large ones. The OUTPUT ENABLE (OE) line **152** from the microprocessor **50** is connected to all memories **150** on the circuit board. The loading on the output enable line **152** to the microprocessor **50** is directly related to the number of memories **150** connected. By monitoring how rapidly OE **152** goes high after a read, the microprocessor **50** is able to determine when the data hold time has been satisfied and place the next address on the bus.

The level of the OE line **152** is monitored by CMOS input buffer **410** which generates an internal READY signal on line **412** to the microprocessor's memory controller. Curves **414** and **416** of the FIG. **15** graph show the difference in rise time likely to be encountered from a lightly to heavily loaded memory system. When the OE line **152** has reached a predetermined level to generate the READY signal, driver **418** generates an OUTPUT ENABLE signal on OE line **152**.

SKIP WITHIN THE INSTRUCTION CACHE

The microprocessor **50** fetches four 8-bit instructions each memory cycle and stores them in a 32-bit instruction register **108**, as shown in FIG. **16**. A class of "test and skip" instructions can very rapidly execute a very fast jump operation within the four instruction cache.

SKIP CONDITIONS:

Always

ACC non-zero

ACC negative

Carry flag equal logic one

Never

ACC equal zero

ACC positive

Carry flag equal logic zero

The SKIP instruction can be located in any of the four byte positions **420** in the 32-bit instruction register **108**. If the test is successful, SKIP will jump over the remaining one, two, or three 8-bit instructions in the instruction register **108** and cause the next four-instruction group to be loaded into the register **108**. As shown, the SKIP operation is implemented by resetting the 2-bit microinstruction counter **180** to zero on line **422** and simultaneously latching the next instruction group into the register **108**. Any instructions following the SKIP in the instruction register are overwritten by the new instructions and not executed.

The advantage of SKIP is that optimizing compilers and smart programmers can often use it in place of the longer conditional JUMP instruction. SKIP also makes possible microloops which exit when the loop counts down or when the SKIP jumps to the next instruction group. The result in very fast code.

Other machines (such as the PDP-8 and Data General NOVA) provide the ability to skip a single instruction. The microprocessor **50** provides the ability to skip up to three instructions.

MICROLOOP IN THE INSTRUCTION CACHE

The microprocessor **50** provides the MICROLOOP instruction to execute repetitively from one to three instructions residing in the instruction register **108**. The microloop instruction works in conjunction with the LOOP COUNTER **92** (FIG. **2**) connected to the internal data bus **90**. To execute a microloop, the program stores a count in LOOP COUNTER **92**. MICROLOOP may be placed in the first, second, third, or last byte **420** of the instruction register **108**. If placed in the first position, execution will just create a delay equal to the number stored in LOOP COUNTER **92** times the machine cycle. If placed in the second, third, or last byte **420**, when the microloop instruction is executed, it will test the LOOP COUNT for zero. If zero, execution will continue with the next instruction. If not zero, the LOOP COUNTER **92** is decremented and the 2-bit microinstruction counter is cleared, causing the preceding instructions in the instruction register to be executed again.

Microloop is useful for block move and search operations. By executing a block move completely out of the instruction register **108**, the speed of the move is doubled, since all memory cycles are used by the move rather than being shared with instruction fetching. Such a hardware implementation of microloops is much faster than conventional software implementation of a comparable function.

OPTIMAL CPU CLOCK SCHEME

The designer of a high speed microprocessor must produce a product which operate over wide temperature ranges, wide voltage swings, and wide variations in semiconductor processing. Temperature, voltage, and process all affect transistor propagation delays. Traditional CPU designs are done so that with the worse case of the three parameters, the circuit will function at the rated clock speed. The result are designs that must be clocked a factor of two slower than their maximum theoretical performance, so they will operate properly in worse case conditions.

The microprocessor **50** uses the technique shown in FIGS. **17–19** to generate the system clock and its required phases. Clock circuit **430** is the familiar "ring oscillator" used to test process performance. The clock is fabricated on the same silicon chip as the rest of the microprocessor **50**.

The ring oscillator frequency is determined by the parameters of temperature, voltage, and process. At room temperature, the frequency will be in the neighborhood of 100 MHZ. At 70 degrees Centigrade, the speed will be 50 MHZ. The ring oscillator **430** is useful as a system clock, with its stages **431** producing phase 0-phase 3 outputs **433** shown in FIG. **19**, because its performance tracks the parameters which similarly affect all other transistors on the same silicon die. By deriving system timing from the ring

oscillator **430**, CPU **70** will always execute at the maximum frequency possible, but never too fast. For example, if the processing of a particular die is not good resulting in slow transistors, the latches and gates on the microprocessor **50** will operate slower than normal. Since the microprocessor **50** ring oscillator clock **430** is made from the same transistors on the same die as the latches and gates, it too will operate slower (oscillating at a lower frequency), providing compensation which allows the rest of the chip's logic to operate properly.

ASYNCHRONOUS/SYNCHRONOUS CPU

Most microprocessors derive all system timing from a single clock. The disadvantage is that different parts of the system can slow all operations. The microprocessor **50** provides a dual-clock scheme as shown in FIG. **17**, with the CPU **70** operating a synchronously to I/O interface **432** forming part of memory controller **118** (FIG. **2**) and the I/O interface **432** operating synchronously with the external world of memory and I/O devices. The CPU **70** executes at the fastest speed possible using the adaptive ring counter clock **430**. Speed may vary by a factor of four depending upon temperature, voltage, and process. The external world must be synchronized to the microprocessor **50** for operations such as video display updating and disc drive reading and writing. This synchronization is performed by the I/O interface **432**, speed of which is controlled by a conventional crystal clock **434**. The interface **432** processes requests for memory accesses from the microprocessor **50** and acknowledges the presence of I/O data. The microprocessor **50** fetches up to four instructions in a single memory cycle and can perform much useful work before requiring another memory access. By decoupling the variable speed of the CPU **70** from the fixed speed of the I/O interface **432**, optimum performance can be achieved by each. Recoupling between the CPU **70** and the interface **432** is accomplished with handshake signals on lines **436**, with data/addresses passing on bus **90**, **136**.

ASYNCHRONOUS/SYNCHRONOUS CPU IMBEDDED ON A DRAM CHIP

System performance is enhanced even more when the DRAM **311** and CPU **314** (FIG. **9**) are located on the same die. The proximity of the transistors means that DRAM **311** and CPU **314** parameters will closely follow each other. At room temperature, not only would the CPU **314** execute at 100 MHZ, but the DRAM **311** would access fast enough to keep up. The synchronization performed by the I/O interface **432** would be for DMA and reading and writing I/O ports. In some systems (such as calculators) no I/O synchronization at all would be required, and the I/O clock would be tied to the ring counter clock.

VARIABLE WIDTH OPERANDS

Many microprocessors provide variable width operands. The microprocessor **50** handles operands of 8, 16, or 24 bits using the same op-code. FIG. **20** shows the 32-bit instruction register **108** and the 2-bit microinstruction register **180** which selects the 8-bit instruction. Two classes of microprocessor **50** instructions can be greater than 8-bits, JUMP class and IMMEDIATE. A JUMP or IMMEDIATE op-code is 8-bits, but the operand can be 8, 16, or 24 bits long. This magic is possible because operands must be right justified in the instruction register. This means that the least significant bit of the operand is always located in the least significant bit of the instruction register. The microinstruction counter **180** selects which 8-bit instruction to execute. If a JUMP or IMMEDIATE instruction is decoded, the state of the 2-bit microinstruction counter selects the required 8, 16, or 24 bit operand onto the address or data bus. The unselected 8-bit bytes are loaded with zeros by operation of decoder **440** and gates **442**. The advantage of this technique is the saving of a number of op-codes required to specify the different operand sizes in other microprocessors.

TRIPLE STACK CACHE

Computer performance is directly related to the system memory bandwidth. The faster the memories, the faster the computer. Fast memories are expensive, so techniques have been developed to move a small amount of high-speed memory around to the memory addresses where it is needed. A large amount of slow memory is constantly updated by the fast memory, giving the appearance of a large fast memory array. A common implementation of the technique is known as a high-speed memory cache. The cache may be thought of as fast acting shock absorber smoothing out the bumps in memory access. When more memory is required than the shock can absorb, it bottoms out and slow speed memory is accessed. Most memory operations can be handled by the shock absorber itself.

The microprocessor **50** architecture has the ALU **80** (FIG. **2**) directly coupled to the top two stack locations **76** and **78**. The access time of the stack **74** therefore directly affects the execution speed of the processor. The microprocessor **50** stack architecture is particularly suitable to a triple cache technique, shown in FIG. **21** which offers the appearance of a large stack memory operating at the speed of on-chip latches **450**. Latches **450** are the fastest form of memory device built on the chip, delivering data in as little as 3 nsec. However latches **450** require large numbers of transistors to construct. On-chip RAM **452** requires fewer transistors than latches, but is slower by a factor of five (15 nsec access). Off-chip RAM **150** is the slowest storage of all. The microprocessor **50** organizes the stack memory hierarchy as three interconnected stacks **450**, **452** and **454**. The latch stack **450** is the fastest and most frequently used. The on-chip RAM stack **452** is next. The off-chip RAM stack **454** is slowest. The stack modulation determines the effective access time of the stack. If a group of stack operations never push or pull more than four consecutive items on the stack, operations will be entirely performed in the 3 nsec latch stack. When the four latches **456** are filled, the data in the bottom of the latch stack **450** is written to the top of the on-chip RAM stack **452**. When the sixteen locations **458** in the on-chip RAM stack **452** are filled, the data in the bottom of the on-chip RAM stack **452** is written to the top of the off-chip RAM stack **454**. When popping data off a full stack **450**, four pops will be performed before stack empty line **460** from the latch stack pointer **462** transfers data from the on-chip RAM stack **452**. By waiting for the latch stack **450** to empty before performing the slower on-chip RAM access, the high effective speed of the latches **456** are made available to the processor. The same approach is employed with the on-chip RAM stack **452** and the off-chip RAM stack **454**.

POLYNOMIAL GENERATION INSTRUCTION

Polynomials are useful for error correction, encryption, data compression, and fractal generation. A polynomial is generated by a sequence of shift and exclusive OR operations. Special chips are provided for this purpose in the prior art.

The microprocessor **50** is able to generate polynomials at high speed without external hardware by slightly modifying how the ALU **80** works. As shown in FIG. **21**, a polynomial is generated by loading the "order" (also known as the feedback terms) into C Register **470**. The value thirty one (resulting in 32 iterations) is loaded into DOWN COUNTER **472**. A register **474** is loaded with zero. B register **476** is loaded with the starting polynomial value. When the POLY

instruction executes, C register **470** is exclusively ORed with A register **474** if the least significant bit of B register **476** is a one. Otherwise, the contents of the A register **474** passes through the ALU **80** unaltered. The combination of A and B is then shifted right (divided by 2) with shifters **478** and **480**. The operation automatically repeats the specified number of iterations, and the resulting polynomial is left in A register **474**.

FAST MULTIPLY

Most microprocessors offer a 16×16 or 32×32 bit multiply instruction. Multiply when performed sequentially takes one shift/add per bit, or 32 cycles for 32 bit data. The microprocessor **50** provides a high speed multiply which allows multiplication by small numbers using only a small number of cycles. FIG. **23** shows the logic used to implement the high speed algorithm. To perform a multiply, the size of the multiplier less one is placed in the DOWN COUNTER **472**. For a four bit multiplier, the number three would be stored in the DOWN COUNTER **472**. Zero is loaded into the A register **474**. The multiplier is written bit reversed into the B Register **476**. For example, a bit reversed five (binary 0101) would be written into B as 1010. The multiplicand is written into the C register **470**. Executing the FAST MULT instruction will leave the result in the A Register **474**, when the count has been completed. The fast multiply instruction is important because many applications scale one number by a much smaller number. The difference in speed between multiplying a 32×32 bit and a 32×4 bit is a factor of 8. If the least significant bit of the multiplier is a "ONE", the contents of the A register **474** and the C register **470** are added. If the least significant bit of the multiplier is a "ZERO", the contents of the A register are passed through the ALU **80** unaltered. The output of the ALU **80** is shifted left by shifter **482** in each iteration. The contents of the B register **476** are shifted right by the shifter **480** in each iteration.

INSTRUCTION EXECUTION PHILOSOPHY

The microprocessor **50** uses high speed D latches in most of the speed critical areas. Slower on-chip RAM is used as secondary storage.

The microprocessor **50** philosophy of instruction execution is to create a hierarchy of speed as follows:

Logic and D latch transfers	1 cycle	20 nsec
Math	2 cycles	40 nsec
Fetch/store on-chip RAM	2 cycles	40 nsec
Fetch/store in current RAS page	4 cycles	80 nsec
Fetch/store with RAS cycle	11 cycles	220 nsec

With a 50 MHZ clock, many operations can be performed in 20 nsec. and almost everything else in 40 nsec.

To maximize speed, certain techniques in processor design have been used. They include:

Eliminating arithmetic operations on addresses,

Fetching up to four instructions per memory cycle,

Pipelineless instruction decoding

Generating results before they are needed,

Use of three level stack caching.

PIPELINE PHILOSOPHY

Computer instructions are usually broken down into sequential pieces, for example: fetch, decode, register read, execute, and store. Each piece will require a single machine cycle. In most Reduced Instruction Set Computer (RISC) chips, instruction require from three to six cycles.

RISC instructions are very parallel. For example, each of 70 different instructions in the SPARC (SUN Computer's RISC chip) has five cycles. Using a technique called "pipelining", the different phases of consecutive instructions can be overlapped.

To understand pipelining, think of building five residential homes. Each home will require in sequence, a foundation, framing, plumbing and wiring, roofing, and interior finish. Assume that each activity takes one week. To build one house will take five weeks.

But what if you want to build an entire subdivision? You have only one of each work crew, but when the foundation men finish on the first house, you immediately start them on the second one, and so on. At the end of five weeks, the first home is complete, but you also have five foundations. If you have kept the framing, plumbing, roofing, and interior guys all busy, from five weeks on, a new house will be completed each week.

This is the way a RISC chip like SPARC appears to execute an instruction in a single machine cycle. In reality, a RISC chip is executing one fifth of five instructions each machine cycle. And if five instructions stay in sequence, an instruction will be completed each machine cycle.

The problems with a pipeline are keeping the pipe full with instructions. Each time an out of sequence instruction such as a BRANCH or CALL occurs, the pipe must be refilled with the next sequence. The resulting dead time to refill the pipeline can become substantial when many IF/THEN/ELSE statements or subroutines are encountered.

THE PIPELINE APPROACH

The microprocessor **50** has no pipeline as such. The approach of this microprocessor to speed is to overlap instruction fetching with execution of the previously fetched instruction(s). Beyond that, over half the instructions (the most common ones) execute entirely in a single machine cycle of 20 nsec. This is possible because:

1. Instruction decoding resolves in 2.5 nsec.
2. Incremented/decremented and some math values are calculated before they are needed, requiring only a latching signal to execute.
3. Slower memory is hidden from high speed operations by high-speed D latches which access in 4 nsec.

The disadvantage for this microprocessor is a more complex chip design process. The advantage for the chip user is faster ultimate throughput since pipeline stalls cannot exist. Pipeline synchronization with availability flag bits and other such pipeline handling is not required by this microprocessor.

For example, in some RISC machines an instruction which tests a status flag may have to wait for up to four cycles for the flag set by the previous instruction to be available to be tested. Hardware and software debugging is also somewhat easier because the user doesn't have to visualize five instructions simultaneously in the pipe.

OVERLAPPING INSTRUCTION FETCH/EXECUTE

The slowest procedure the microprocessor **50** performs is to access memory. Memory is accessed when data is read or written. Memory is also read when instructions are fetched. The microprocessor **50** is able to hide fetch of the next instruction behind the execution of the previously fetched instruction(s). The microprocessor **50** fetches instructions in 4-byte instruction groups. An instruction group may contain from one to four instructions. The amount of time required to execute the instruction group ranges from 4 cycles for simple instructions to 64 cycles for a multiply.

When a new instruction group is fetched, the microprocessor instruction decoder looks at the most significant bit of all four of the bytes. The most significant bit of an instruction determines if a memory access is required. For example, CALL, FETCH, and STORE all require a memory access to

execute. If all four bytes have nonzero most significant bits, the microprocessor initiates the memory fetch of the next sequential 4-byte instruction group. When the last instruction in the group finishes executing, the next 4-byte instruction group is ready and waiting on the data bus needing only to be latched into the instruction register. If the 4-byte instruction group required four or more cycles to execute and the next sequential access was a column address strobe (CAS) cycle, the instruction fetch was completely overlapped with execution.

INTERNAL ARCHITECTURE

The microprocessor **50** architecture consists of the following:

PARAMETER STACK	<---> ALU* <--->	Y REGISTER RETURN STACK
<---32 BITS---> 16 DEEP Used for math and logic.		<---32 BITS---> 16 DEEP Used for subroutine and interrupt return addresses as well as local variables.
Push down stack. Can overflow into off-chip RAM.		Push down stack. Can overflow into off-chip RAM. Can also be accessed relative to top of stack.
LOOP COUNTER	(32-bits, can decrement by 1) Used by class of test and loop instructions.	
X REGISTER	(32-bits, can increment or decrement by 4). Used to point to RAM locations.	
PROGRAM COUNTER	(32-bits, increments by 4). Points to 4-byte instruction groups in RAM.	
INSTRUCTION REG	(32-Bits). Holds 4-byte instruction groups while they are being decoded and executed.	

MODE - A register with mode and status bits.

MODE-BITS:

- Slow down memory accesses by 8 if "1". Run full speed if "0". (Provided for access to slow EPROM.)
- Divide the system clock by 1023 if "1" to reduce power consumption. Run full speed if "0". (On-chip counters slow down if this bit is set.)
- Enable external interrupt 1.
- Enable external interrupt 2.
- Enable external interrupt 3.
- Enable external interrupt 4.
- Enable external interrupt 5.
- Enable external interrupt 6.
- Enable external interrupt 7.

ON-CHIP MEMORY LOCATIONS:

MODE-BITS	
DMA-POINTER	
DMA-COUNTER	
STACK-POINTER	- Pointer into Parameter Stack.
STACK-DEPTH	- Depth of on-chip Parameter Stack
RSTACK-POINTER	- Pointer into Return Stack
RSTACK-DEPTH	- Depth of on-chip Return Stack

*Math and logic operations use the TOP item and NEXT to top Parameter Stack items as the operands. The result is pushed onto the Parameter Stack. *Return addresses from subroutines are placed on the Return Stack. The Y REGISTER is used as a pointer to RAM locations. Since the Y REGISTER is the top item of the Return Stack, nesting of indices is straightforward.

ADDRESSING MODE HIGH POINTS

The data bus is 32-bits wide. All memory fetches and stores are 32-bits. Memory bus addresses are 30 bits. The least significant 2 bits are used to select one-of-four bytes in some addressing modes. The Program Counter, X Register, and Y Register are implemented as D latches with their outputs going to the memory address bus and the bus incrementer/decrementer. Incrementing one of these registers can happen quickly, because the incremented value has already rippled through the inc/dec logic and need only be clocked into the latch. Branches and Calls are made to 32-bit word boundaries.

INSTRUCTION SET

32-BIT INSTRUCTION FORMAT

The thirty two bit instructions are CALL, BRANCH, BRANCH-IF-ZERO, and LOOP-IF-NOT-DONE. These instructions require the calculation of an effective address. In many computers, the effective address is calculated by adding or subtracting an operand with the current Program Counter. This math operation requires from four to seven machine cycles to perform and can definitely bog down machine execution. The microprocessor's strategy is to perform the required math operation at assembly or linking time and do a much simpler "Increment to next page" or "Decrement to previous page" operation at run time. As a result, the microprocessor branches execute in a single cycle.

24-BIT OPERAND FORM:

Byte 1	Byte 2	Byte 3	Byte 4
WWWWWW XX -	YYYYYYYY -	YYYYYYYY -	YYYYYYYY

With a 24-bit operand, the current page is considered to be defined by the most significant 6 bits of the Program Counter.

16-BIT OPERAND FORM: QQQQQQQQ-WWWWWW XX-YYYYYYYY-YYYYYYYY With a 16-bit operand, the current page is considered to be defined by the most significant 14 bits of the Program Counter.

8-BIT OPERAND FORM: QQQQQQQQ-QQQQQQQQ-WWWWWW XX-YYYYYYYY With an 8-bit operand, the current page is considered to be defined by the most significant 22 bits of the Program Counter.

QQQQQQQQ—Any 8-bit instruction.

WWWWWW—Instruction op-code.

XX—Select how the address bits will be used:

00—Make all high-order bits zero. (Page zero addressing)

01—Increment the high-order bits. (Use next page)

10—Decrement the high-order bits. (Use previous page)

11—Leave the high-order bits unchanged. (Use current page)

YYYYYYYY—The address operand field. This field is always shifted left two bits (to generate a word rather than byte address) and loaded into the Program Counter. The microprocessor instruction decoder figures out the width of the operand field by the location of the instruction op-code in the four bytes.

The compiler or assembler will normally use the shortest operand required to reach the desired address so that the leading bytes can be used to hold other instructions. The effective address is calculated by combining:

The current Program Counter,

The 8, 16, or 24 bit address operand in the instruction,

Using one of the four allowed addressing modes.

EXAMPLES OF EFFECTIVE ADDRESS CALCULATION

Example 1

Byte 1	Byte 2	Byte 3	Byte 4
QQQQQQQQ	QQQQQQQQ	00000011	10011000

The "QQQQQQQQs" in Byte 1 and 2 indicate space in the 4-byte memory fetch which could be hold two other

instructions to be executed prior to the CALL instruction. Byte 3 indicates a CALL instruction (six zeros) in the current page (indicated by the 11 bits). Byte 4 indicates that the hexadecimal number **98** will be forced into the Program Counter bits **2** through **10**. (Remember, a CALL or BRANCH always goes to a word boundary so the two least significant bits are always set to zero). The effect of this instruction would be to CALL a subroutine at WORD location HEX 98 in the current page. The most significant 22 bits of the Program Counter define the current page and will be unchanged.

Example 2

Byte 1	Byte 2	Byte 3	Byte 4
000001 01	00000001	00000000	00000000

If we assume that the Program Counter was HEX 0000 0156 which is binary:

00000000 00000000 00000001 01010110=OLD PROGRAM COUNTER.

Byte 1 indicates a BRANCH instruction op code (000001) and "01" indicates select the next page. Byte 2,3, and 4 are the address operand. These 24-bits will be shifted to the left two places to define a WORD address. HEX 0156 shifted left two places is HEX 0558. Since this is a 24-bit operand instruction, the most significant 6 bits of the Program Counter define the current page. These six bits will be incremented to select the next page. Executing this instruction will cause the Program Counter to be loaded with HEX 0400 0558 which is binary:

00000100 00000000 00000101 01011000=NEW PROGRAM COUNTER.

INSTRUCTIONS

CALL-LONG

0000 00XX-YYYYYYYY-YYYYYYYY-YYYYYYYY

Load the Program Counter with the effective WORD address specified. Push the current PC contents onto the RETURN STACK.

OTHER EFFECTS: CARRY or modes, no effect. May cause Return Stack to force an external memory cycle if on-chip Return Stack is full.

BRANCH

0000 01XX-YYYYYYYY-YYYYYYYY-YYYYYYYY

Load the Program Counter with the effective WORD address specified.

OTHER EFFECTS: NONE

BRANCH-IF-ZERO

0000 10XX-YYYYYYYY-YYYYYYYY-YYYYYYYY

Test the TOP value on the Parameter Stack. If the value is equal to zero, load the Program Counter with the effective WORD address specified. If the TOP value is not equal to zero, increment the Program Counter and fetch and execute the next instruction.

OTHER EFFECTS: NONE

LOOP-IF-NOT-DONE

0000 11YY-(XXXX XXXX)-(XXXX XXXX)-(XXXX XXXX)

If the LOOP COUNTER is not zero, load the Program Counter with the effective WORD address specified. If the LOOP COUNTER is zero, decrement the LOOP COUNTER, increment the Program Counter and fetch and execute the next instruction.

OTHER EFFECTS: NONE

8-BIT INSTRUCTIONS PHILOSOPHY

Most of the work in the microprocessor **50** is done by the 8-bit instructions. Eight bit instructions are possible with the microprocessor because of the extensive use of implied stack addressing. Many 32-bit architectures use 8-bits to specify the operation to perform but use an additional 24-bits to specify two sources and a destination.

For math and logic operations, the microprocessor **50** exploits the inherent advantage of a stack by designating the source operand(s) as the top stack item and the next stack item. The math or logic operation is performed, the operands are popped from the stack, and the result is pushed back on the stack. The result is a very efficient utilization of instruction bits as well as registers. A comparable situation exists between Hewlett Packard calculators (which use a stack) and Texas Instrument calculators which don't. The identical operation on an HP will require one half to one third the keystrokes of the TI.

The availability of 8-bit instructions also allows another architectural innovation, the fetching of four instructions in a single 32-bit memory cycle. The advantages of fetching multiple instructions are:

Increased execution speed even with slow memories,

Similar performance to the Harvard (separate data and instruction busses) without the expense,

Opportunities to optimize groups of instructions,

The capability to perform loops within this mini-cache. The microloops inside the four instruction group are effective for searches and block moves.

SKIP INSTRUCTIONS

The microprocessor **50** fetches instructions in 32-bit chunks called 4-byte instruction groups. These four bytes may contain four 8-bit instructions or some mix of 8-bit and 16 or 24-bit instructions. SKIP instructions in the microprocessor skip any remaining instructions in a 4-byte instruction group and cause a memory fetch to get the next 4-byte instruction group. Conditional SKIPs when combined with 3-byte BRANCHES will create conditional BRANCHES. SKIPs may also be used in situations when no use can be made of the remaining bytes in a 4-instruction group. A SKIP executes in a single cycle, whereas a group of three NOPs would take three cycles.

SKIP-ALWAYS—Skip any remaining instructions in this 4-byte instruction group. Increment the most significant 30-bits of the Program Counter and proceed to fetch the next 4-byte instruction group.

SKIP-IF-ZERO—If the TOP item of the Parameter Stack is zero, skip any remaining instructions in the 4-byte instruction group. Increment the most significant 30-bits of the Program Counter and proceed to fetch the next 4-byte instruction group. If the TOP item is not zero, execute the next sequential instruction.

SKIP-IF-POSITIVE—If the TOP item of the Parameter Stack has a the most significant bit (the sign bit) equal to "0", skip any remaining instructions in the 4-byte instruction group. Increment the most significant 30-bits of the Program Counter and proceed to fetch the next 4-byte instruction group. If the TOP item is not "0", execute the next sequential instruction.

SKIP-IF-NO-CARRY—If the CARRY flag from a SHIFT or arithmetic operation is not equal to "1", skip any remaining instructions in the 4-byte instruction group. Increment the most significant 30-bits of the Program Counter and proceed to fetch the next 4-byte instruction group. If the CARRY is equal to "1", execute the next sequential instruction.

SKIP-NEVER (NOP) execute the next sequential instruction. (Delay one machine cycle).

SKIP-IF-NOT-ZERO—If the TOP item on the Parameter Stack is not equal to "0", skip any remaining instructions

in the 4-byte instruction group. Increment the most significant 30-bits of the Program Counter and proceed to fetch the next 4-byte instruction group. If the TOP item is equal "0", execute the next sequential instruction.

SKIP-IF-NEGATIVE—If the TOP item on the Parameter Stack has its most significant bit (sign bit) set to "1", skip any remaining instructions in the 4-byte instruction group. Increment the most significant 30-bits of the Program Counter and proceed to fetch the next 4-byte instruction group. If the TOP item has its most significant bit set to "0", execute the next sequential instruction.

SKIP-IF-CARRY—If the CARRY flag is set to "1" as a result of SHIFT or arithmetic operation, skip any remaining instructions in the 4-byte instruction group. Increment the most significant 30-bits of the Program Counter and proceed to fetch the next 4-byte instruction group. If the CARRY flag is "0", execute the next sequential instruction.

MICROLOOPS

Microloops are a unique feature of the microprocessor architecture which allows controlled looping within a 4-byte instruction group. A microloop instruction tests the LOOP COUNTER for "0" and may perform an additional test. If the LOOP COUNTER is not "0" and the test is met, instruction execution continues with the first instruction in the 4-byte instruction group, and the LOOP COUNTER is decremented. A microloop instruction will usually be the last byte in a 4-byte instruction group, but it can be any byte. If the LOOP COUNTER is "0" or the test is not met, instruction execution continues with the next instruction. If the microloop is the last byte in the 4-byte instruction group, the most significant 30-bits of the Program Counter are incremented and the next 4-byte instruction group is fetched from memory. On a termination of the loop on LOOP COUNTER equal to "0", the LOOP COUNTER will remain at "0". Microloops allow short iterative work such as moves and searches to be performed without slowing down to fetch instructions from memory.

EXAMPLE

Byte 1 FETCH-VIA-X-AUTO-INCREMENT	Byte 2 STORE-VIA-Y-AUTOINCREMENT
Byte 3 ULOOP-UNTIL-DONE	Byte 4 QQQQQQQQ

This example will perform a block move. To initiate the transfer, X will be loaded with the starting address of the source. Y will be loaded with the starting address of the destination. The LOOP COUNTER will be loaded with the number of 32-bit words to move. The microloop will FETCH and STORE and count down the LOOP COUNTER until it reaches zero. QQQQQQQQ indicates any instruction can follow.

MICROLOOP INSTRUCTIONS

ULOOP-UNTIL-DONE—If the LOOP COUNTER is not "0", continue execution with the first instruction in the 4-byte instruction group. Decrement the LOOP COUNTER. If the LOOP COUNTER is "0", continue execution with the next instruction.

ULOOP-IF-ZERO—If the LOOP COUNTER is not "0" and the TOP item on the Parameter Stack is "0", continue execution with the first instruction in the 4-byte instruction group. Decrement the LOOP COUNTER. If the LOOP COUNTER is "0" or the TOP item is "1", continue execution with the next instruction.

ULOOP-IF-POSITIVE—If the LOOP COUNTER is not "0" and the most significant bit (sign bit) is "0", continue execution with the first instruction in the 4-byte instruction group. Decrement the LOOP COUNTER. If the LOOP COUNTER is "0" or the TOP item is "1", continue execution with the next instruction.

ULOOP-IF-NOT-CARRY-CLEAR—If the LOOP COUNTER is not "0" and the floating point exponents found in TOP and NEXT are not aligned, continue execution with the first instruction in the 4-byte instruction group. Decrement the LOOP COUNTER. If the LOOP COUNTER is "0" or the exponents are aligned, continue execution with the next instruction. This instruction is specifically designed for combination with special SHIFT instructions to align two floating point numbers.

ULOOP-NEVER—(DECREMENT-LOOP-COUNTER) Decrement the LOOP COUNTER. Continue execution with the next instruction.

ULOOP-IF-NOT-ZERO—If the LOOP COUNTER is not "0" and the TOP item of the Parameter Stack is "0", continue execution with the first instruction in the 4-byte instruction group. Decrement the LOOP COUNTER. If the LOOP COUNTER is "0" or the TOP item is "1", continue execution with the next instruction.

ULOOP-IF-NEGATIVE—If the LOOP COUNTER is not "0" and the most significant bit (sign bit) of the TOP item of the Parameter Stack is "1", continue execution with the first instruction in the 4-byte instruction group. Decrement the LOOP COUNTER. If the LOOP COUNTER is "0" or the most significant bit of the Parameter Stack is "0", continue execution with the next instruction.

ULOOP-IF-CARRY-SET—If the LOOP COUNTER is not "0" and the exponents of the floating point numbers found in TOP and NEXT are not aligned, continue execution with the first instruction in the 4-byte instruction group. Decrement the LOOP COUNTER. If the LOOP COUNTER is "0" or the exponents are aligned, continue execution with the next instruction.

RETURN FROM SUBROUTINE OR INTERRUPT

Subroutine calls and interrupt acknowledgements cause a redirection of normal program execution. In both cases, the current Program Counter is pushed onto the Return Stack, so the microprocessor can return to its place in the program after executing the subroutine or interrupt service routine.

NOTE: When a CALL to subroutine or interrupt is acknowledged the Program Counter has already been incremented and is pointing to the 4-byte instruction group following the 4-byte group currently being executed. The instruction decoding logic allows the microprocessor to perform a test and execute a return conditional on the outcome of the test in a single cycle. A RETURN pops an address from the Return Stack and stores it to the Program Counter.

RETURN INSTRUCTIONS

RETURN-ALWAYS—Pop the top item from the Return Stack and transfer it to the Program Counter.

RETURN-IF-ZERO—If the TOP item on the Parameter Stack is "0", pop the top item from the Return Stack and transfer it to the Program Counter. Otherwise execute the next instruction.

RETURN-IF-POSITIVE—If the most significant bit (sign bit) of the TOP item on the Parameter Stack is a "0", pop the top item from the Return Stack and transfer it to the Program Counter. Otherwise execute the next instruction.

RETURN-IF-CARRY-CLEAR—If the exponents of the floating point numbers found in TOP and NEXT are not aligned, pop the top item from the Return Stack and

transfer it to the Program Counter. Otherwise execute the next instruction.

RETURN-NEVER (NOP)—Execute the next instruction.

RETURN-IF-NOT-ZERO—If the TOP item on the Parameter Stack is not "0", pop the top item from the Return Stack and transfer it to the Program Counter. Otherwise execute the next instruction.

RETURN-IF-NEGATIVE—If the most significant bit (sign bit) of the TOP item on the Parameter Stack is a "1", pop the top item from the Return Stack and transfer it to the Program Counter. Otherwise execute the next instruction.

RETURN-IF-CARRY-SET—If the exponents of the floating point numbers found in TOP and NEXT are aligned, pop the top item from the Return Stack and transfer it to the Program Counter. Otherwise execute the next instruction.

HANDLING MEMORY FROM DYNAMIC RAM

The microprocessor **50**, like any RISC type architecture, is optimized to handle as many operations as possible on-chip for maximum speed. External memory operations take from 80 nsec. to 220 nsec. compared with on-chip memory speeds of from 4 nsec. to 30 nsec. There are times when external memory must be accessed.

External memory is accessed using three registers:

X-REGISTER—A 30-bit memory pointer which can be used for memory access and simultaneously incremented or decremented.

Y-REGISTER—A 30-bit memory pointer which can be used for memory access and simultaneously incremented or decremented.

PROGRAM-COUNTER—A 30-bit memory pointer normally used to point to 4-byte instruction groups. External memory may be accessed at addresses relative to the PC. The operands are sometimes called "Immediate" or "Literal" in other computers. When used as memory pointer, the PC is also incremented after each operation.

MEMORY LOAD & STORE INSTRUCTIONS

FETCH-VIA-X—Fetch the 32-bit memory content pointed to by X and push it onto the Parameter Stack. X is unchanged.

FETCH-VIA-Y—Fetch the 32-bit memory content pointed to by X and push it onto the Parameter Stack. Y is unchanged.

FETCH-VIA-X-AUTOINCREMENT—Fetch the 32-bit memory content pointed to by X and push it onto the Parameter Stack. After fetching, increment the most significant 30 bits of X to point to the next 32-bit word address.

FETCH-VIA-Y-AUTOINCREMENT—Fetch the 32-bit memory content pointed to by Y and push it onto the Parameter Stack. After fetching, increment the most significant 30 bits of Y to point to the next 32-bit word address.

FETCH-VIA-X-AUTODECREMENT—Fetch the 32-bit memory content pointed to by X and push it onto the Parameter Stack. After fetching, decrement the most significant 30 bits of X to point to the previous 32-bit word address.

FETCH-VIA-Y-AUTODECREMENT—Fetch the 32-bit memory content pointed to by Y and push it onto the Parameter Stack. After fetching, decrement the most significant 30 bits of Y to point to the previous 32-bit word address.

STORE-VIA-X—Pop the top item of the Parameter Stack and store it in the memory location pointed to by X. X is unchanged.

STORE-VIA-Y—Pop the top item of the Parameter Stack and store it in the memory location pointed to by Y. Y is unchanged.

STORE-VIA-X-AUTOINCREMENT—Pop the top item of the Parameter Stack and store it in the memory location pointed to by X. After storing, increment the most significant 30 bits of X to point to the next 32-bit word address.

STORE-VIA-Y-AUTOINCREMENT—Pop the top item of the Parameter Stack and store it in the memory location pointed to by Y. After storing, increment the most significant 30 bits of Y to point to the next 32-bit word address.

STORE-VIA-X-AUTODECREMENT—Pop the top item of the Parameter Stack and store it in the memory location pointed to by X. After storing, decrement the most significant 30 bits of X to point to the previous 32-bit word address.

STORE-VIA-Y-AUTODECREMENT—Pop the top item of the Parameter Stack and store it in the memory location pointed to by Y. After storing, decrement the most significant 30 bits of Y to point to the previous 32-bit word address.

FETCH-VIA-PC—Fetch the 32-bit memory content pointed to by the Program Counter and push it onto the Parameter Stack. After fetching, increment the most significant 30 bits of the Program Counter to point to the next 32-bit word address.

*NOTE When this instruction executes, the PC is pointing to the memory location following the instruction. The effect is of loading a 32-bit immediate operand. This is an 8-bit instruction and therefore will be combined with other 8-bit instructions in a 4-byte instruction fetch. It is possible to have from one to four FETCH-VIA-PC instructions in a 4-byte instruction fetch. The PC increments after each execution of FETCH-VIA-PC, so it is possible to push four immediate operands on the stack. The four operands would be the found in the four memory locations following the instruction.

BYTE-FETCH-VIA-X—Fetch the 32-bit memory content pointed to by the most significant 30 bits of X. Using the two least significant bits of X, select one of four bytes from the 32-bit memory fetch, right justify the byte in a 32-bit field and push the selected byte preceded by leading zeros onto the Parameter Stack.

BYTE-STORE-VIA-X—Fetch the 32-bit memory content pointed to by the most significant 30 bits of X. Pop the TOP item from the Parameter Stack. Using the two least significant bits of X place the least significant byte into the 32-bit memory data and write the 32-bit entity back to the location pointed to by the most significant 30 bits of X.

OTHER EFFECTS OF MEMORY ACCESS INSTRUCTIONS:

Any FETCH instruction will push a value on the Parameter Stack **74**. If the on-chip stack is full, the stack will overflow into off-chip memory stack resulting in an additional memory cycle. Any STORE instruction will pop a value from the Parameter Stack **74**. If the on-chip stack is empty, a memory cycle will be generated to fetch a value from off-chip memory stack.

HANDLING ON-CHIP VARIABLES

High-level languages often allow the creation of LOCAL VARIABLES. These variables are used by a particular procedure and discarded. In cases of nested procedures, layers of these variables must be maintained. On-chip storage is up to five times faster than off-chip RAM, so a means of keeping local variables on-chip can make operations run faster. The microprocessor **50** provides the capability for both on-chip storage of local variables and nesting of multiple levels of variables through the Return Stack.

The Return Stack **134** is implemented as 16 on-chip RAM locations. The most common use for the Return Stack **134** is storage of return addresses from subroutines and interrupt calls. The microprocessor allows these 16 locations to also be used as addressable registers. The 16 locations may be read and written by two instructions which indicate a Return Stack relative address from 0–15. When high-level procedures are nested, the current procedure variables push the previous procedure variables further down the Return Stack **134**. Eventually, the Return Stack will automatically overflow into off-chip RAM.

ON-CHIP VARIABLE INSTRUCTIONS

READ-LOCAL-VARIABLE XXXX—Read the XXXXth location relative to the top of the Return Stack. (XXXX is a binary number from 0000–1111). Push the item read onto the Parameter Stack.

OTHER EFFECTS: If the Parameter Stack is full, the push operation will cause a memory cycle to be generated as one item of the stack is automatically stored to external RAM. The logic which selects the location performs a modulo **16** subtraction. If four local variables have been pushed onto the Return Stack, and an instruction attempts to READ the fifth item, unknown data will be returned.

WRITE-LOCAL-VARIABLE XXXX—Pop the TOP item of the Parameter Stack and write it into the XXXXth location relative to the top of the Return Stack. (XXXX is a binary number from 0000–1111.)

OTHER EFFECTS: If the Parameter Stack is empty, the pop operation will cause a memory cycle to be generated to fetch the Parameter Stack item from external RAM. The logic which selects the location performs a modulo **16** subtraction. If four local variables have been pushed onto the Return Stack, and an instruction attempts to WRITE to the fifth item, it is possible to clobber return addresses or wreak other havoc.

REGISTER AND FLIP-FLOP TRANSFER AND PUSH INSTRUCTIONS

DROP—Pop the TOP item from the Parameter Stack and discard it.

SWAP—Exchange the data in the TOP Parameter Stack location with the data in the NEXT Parameter Stack location.

DUP—Duplicate the TOP item on the Parameter Stack and push it onto the Parameter Stack.

PUSH-LOOP-COUNTER—Push the value in LOOP COUNTER onto the Parameter Stack.

POP-RSTACK-PUSH-TO-STACK—Pop the top item from the Return Stack and push it onto the Parameter Stack.

PUSH-X-REG—Push the value in the X Register onto the Parameter Stack.

PUSH-STACK-POINTER—Push the value of the Parameter Stack pointer onto the Parameter Stack.

PUSH-RSTACK-POINTER—Push the value of the Return Stack pointer onto the Return Stack.

PUSH-MODE-BITS—Push the value of the MODE REGISTER onto the Parameter Stack.

PUSH-INPUT—Read the 10 dedicated input bits and push the value (right justified and padded with leading zeros) onto the Parameter Stack.

SET-LOOP-COUNTER—Pop the TOP value from the Parameter Stack and store it into LOOP COUNTER.

POP-STACK-PUSH-TO-RSTACK—Pop the TOP item from the Parameter Stack and push it onto the Return Stack.

SET-X-REG—Pop the TOP item from the Parameter Stack and store it into the X Register.

SET-STACK-POINTER—Pop the TOP item from the Parameter Stack and store it into the Stack Pointer.

SET-RSTACK-POINTER—Pop the TOP item from the Parameter

Stack and store it into the Return Stack Pointer.

SET-MODE-BITS—Pop the TOP value from the Parameter Stack and store it into the MODE BITS.

SET-OUTPUT—Pop the TOP item from the Parameter Stack and output it to the 10 dedicated output bits.

OTHER EFFECTS: Instructions which push or pop the Parameter Stack or Return Stack may cause a memory cycle as the stacks overflow back and forth between on-chip and off-chip memory.

LOADING A SHORT LITERAL

A special case of register transfer instruction is used to push an 8-bit literal onto the Parameter Stack. This instruction requires that the 8-bits to be pushed reside in the last byte of a 4-byte instruction group. The instruction op-code loading the literal may reside in ANY of the other three bytes in the instruction group.

EXAMPLE

BYTE 1	BYTE 2	BYTE 3
LOAD-SHORT-LITERAL	QQQQQQQQ	QQQQQQQQ
BYTE 4		
00001111		

In this example, QQQQQQQQ indicates any other 8-bit instruction. When Byte 1 is executed, binary 00001111(HEX 0f) from Byte 4 will be pushed (right justified and padded by leading zeros) onto the Parameter Stack. Then the instructions in Byte 2 and Byte 3 will execute. The microprocessor instruction decoder knows not to execute Byte 4. It is possible to push three identical 8-bit values as follows:

BYTE 1	BYTE 2
LOAD-SHORT-LITERAL	LOAD-SHORT-LITERAL
BYTE 3	BYTE 4
LOAD-SHORT-LITERAL	00001111
SHORT-LITERAL-INSTRUCTION	

LOAD-SHORT-LITERAL—Push the 8-bit value found in Byte 4 of the current 4-byte instruction group onto the Parameter Stack.

LOGIC INSTRUCTIONS

Logical and math operations used the stack for the source of one or two operands and as the destination for results. The stack organization is a particularly convenient arrangement for evaluating expressions. TOP indicates the top value on the Parameter Stack **74**. NEXT indicates the next to top value on the Parameter Stack **74**.

AND—Pop TOP and NEXT from the Parameter Stack, perform the logical AND operation on these two operands, and push the result onto the Parameter Stack.

OR—Pop TOP and NEXT from the Parameter Stack, perform the logical OR operation on these two operands, and push the result onto the Parameter Stack.

XOR—Pop TOP and NEXT from the Parameter Stack, perform the logical exclusive OR on these two operands, and push the result onto the Parameter Stack.

BIT-CLEAR—Pop TOP and NEXT from the Parameter Stack, toggle all bits in NEXT, perform the logical AND operation on TOP, and push the result onto the Parameter Stack. (Another way of understanding this instruction is thinking of it as clearing all bits in TOP that are set in NEXT.)

MATH INSTRUCTIONS

Math instruction pop the TOP item and NEXT to top item of the Parameter Stack **74** to use as the operands. The results are pushed back on the Parameter Stack. The CARRY flag is used to latch the "33rd bit" of the ALU result.

ADD—Pop the TOP item and NEXT to top item from the Parameter Stack, add the values together and push the result back on the Parameter Stack. The CARRY flag may be changed.

ADD-WITH-CARRY—Pop the TOP item and the NEXT to top item from the Parameter Stack, add the values together. If the CARRY flag is "1" increment the result. Push the ultimate result back on the Parameter Stack. The CARRY flag may be changed.

ADD-X—Pop the TOP item from the Parameter Stack and read the third item from the top of the Parameter Stack. Add the values together and push the result back on the Parameter Stack. The CARRY flag may be changed.

SUB—Pop the TOP item and NEXT to top item from the Parameter Stack, Subtract NEXT from TOP and push the result back on the Parameter Stack. The CARRY flag may be changed.

SUB-WITH-CARRY—Pop the TOP item and NEXT to top item from the Parameter Stack. Subtract NEXT from TOP. If the CARRY flag is "1" increment the result. Push the ultimate result back on the Parameter Stack. The CARRY flag may be changed.

SUB-X—

SIGNED-MULT-STEP—

UNSIGNED-MULT-STEP—

SIGNED-FAST-MULT—

FAST-MULT-STEP—

UNSIGNED-DIV-STEP—

GENERATE-POLYNOMIAL—

ROUND—

COMPARE—Pop the TOP item and NEXT to top item from the Parameter Stack. Subtract NEXT from TOP. If the result has the most significant bit equal to "0" (the result is positive), push the result onto the Parameter Stack. If the result has the most significant bit equal to "1" (the result is negative), push the old value of TOP onto the Parameter Stack. The CARRY flag may be affected.

SHIFT/ROTATE

SHIFT-LEFT—Shift the TOP Parameter Stack item left one bit. The CARRY flag is shifted into the least significant bit of TOP.

SHIFT-RIGHT—Shift the TOP Parameter Stack item right one bit. The least significant bit of TOP is shifted into the CARRY flag. Zero is shifted into the most significant bit of TOP.

DOUBLE-SHIFT-LEFT—Treating the TOP item of the Parameter Stack as the most significant word of a 64-bit number and the NEXT stack item as the least significant word, shift the combined 64-bit entity left one bit. The CARRY flag is shifted into the least significant bit of NEXT.

DOUBLE-SHIFT-RIGHT—Treating the TOP item of the Parameter Stack as the most significant word of a 64-bit number and the NEXT stack item as the least significant word, shift the combined 64-bit entity right one bit. The least significant bit of NEXT is shifted into the CARRY flag. Zero is shifted into the most significant bit of TOP.

OTHER INSTRUCTIONS

FLUSH-STACK—Empty all on-chip Parameter Stack locations into off-chip RAM. (This instruction is useful for multitasking applications). This instruction accesses a counter which holds the depth of the on-chip stack and can require from none to 16 external memory cycles.

FLUSH-RSTACK—Empty all on-chip Return Stack locations into off-chip RAM. (This instruction is useful for multitasking applications). This instruction accesses a counter which holds the depth of the on-chip Return Stack and can require from none to 16 external memory cycles.

It should further be apparent to those skilled in the art that various changes in form and details of the invention as shown and described may be made. It is intended that such changes be included within the spirit and scope of the claims appended hereto.

What is claimed is:

1. A microprocessor system, comprising a single integrated circuit including a central processing unit and an entire ring oscillator variable speed system clock in said single integrated circuit and connected to said central processing unit for clocking said central processing unit, said central processing unit and said ring oscillator variable speed system clock each including a plurality of electronic devices correspondingly constructed of the same process technology with corresponding manufacturing variations, a processing frequency capability of said central processing unit and a speed of said ring oscillator variable speed system clock varying together due to said manufacturing variations and due to at least operating voltage and temperature of said single integrated circuit; an on-chip input/output interface connected to exchange coupling control signals, addresses and data with said central processing unit; and a second clock independent of said ring oscillator variable speed system clock connected to said input/output interface.

2. The microprocessor system of claim **1** in which said second clock is a fixed frequency clock.

3. In a microprocessor integrated circuit, a method for clocking the microprocessor within the integrated circuit, comprising the steps of:

providing an entire ring oscillator system clock constructed of electronic devices within the integrated circuit, said electronic devices having operating characteristics which will, because said entire ring oscillator system clock and said microprocessor are located within the same integrated circuit, vary together with operating characteristics of electronic devices included within the microprocessor;

using the ring oscillator system clock for clocking the microprocessor, said microprocessor operating at a variable processing frequency dependent upon a variable speed of said ring oscillator system clock;

providing an on chip input/output interface for the microprocessor integrated circuit; and

clocking the input/output interface with a second clock independent of the ring oscillator system clock.

4. The method of claim **3** in which the second clock is a fixed frequency clock.

5. The method of claim **3** further including the step of:

transferring information to and from said microprocessor in synchrony with said ring oscillator system clock.

6. A microprocessor system comprising:

a central processing unit disposed upon an integrated circuit substrate, said central processing unit operating at a processing frequency and being constructed of a first plurality of electronic devices;

an entire oscillator disposed upon said integrated circuit substrate and connected to said central processing unit, said oscillator clocking said central processing unit at a clock rate and being constructed of a second plurality of electronic devices, thus varying the processing frequency of said first plurality of electronic devices and

the clock rate of said second plurality of electronic devices in the same way as a function of parameter variation in one or more fabrication or operational parameters associated with said integrated circuit substrate, thereby enabling said processing frequency to track said clock rate in response to said parameter variation;

an on-chip input/output interface, connected between said said central processing unit and an external memory bus, for facilitating exchanging coupling control signals, addresses and data with said central processing unit; and

an external clock, independent of said oscillator, connected to said input/output interface wherein said external clock is operative at a frequency independent of a clock frequency of said oscillator.

7. The microprocessor system of claim **6** wherein said one or more operational parameters include operating temperature of said substrate or operating voltage of said substrate.

8. The microprocessor system of claim **6** wherein said external clock comprises a fixed-frequency clock which operates synchronously relative to said oscillator.

9. The microprocessor system of claim **6** wherein said oscillator comprises a ring oscillator.

10. In a microprocessor system including a central processing unit, a method for clocking said central processing unit comprising the steps of:

providing said central processing unit upon an integrated circuit substrate, said central processing unit being constructed of a first plurality of transistors and being operative at a processing frequency;

providing an entire variable speed clock disposed upon said integrated circuit substrate, said variable speed clock being constructed of a second plurality of transistors;

clocking said central processing unit at a clock rate using said variable speed clock with said central processing unit being clocked by said variable speed clock at a variable frequency dependent upon variation in one or more fabrication or operational parameters associated with said integrated circuit substrate, said processing frequency and said clock rate varying in the same way relative to said variation in said one or more fabrication or operational parameters associated with said integrated circuit substrate;

connecting an on chip input/output interface between said central processing unit and an external memory bus, and exchanging coupling control signals, addresses and data between said input/output interface and said central processing unit; and

clocking said input/output interface using an external clock wherein said external clock is operative at a frequency independent of a clock frequency of said oscillator.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE

CERTIFICATE OF CORRECTION

PATENT NO. : 5,809,336
APPLICATION NO. : 08/484918
DATED : September 15, 1998
INVENTOR(S) : Moore et al.

It is certified that error appears in the above-identified patent and that said Letters Patent is hereby corrected as shown below:

Column 34,
Line 25, delete "oscillator" and insert --variable speed clock--.

Signed and Sealed this

Twenty-second Day of May, 2007

JON W. DUDAS
Director of the United States Patent and Trademark Office

US005809336C1

(12) **EX PARTE REEXAMINATION CERTIFICATE** (7235th)

United States Patent
Moore et al.

(10) **Number: US 5,809,336 C1**
(45) **Certificate Issued: Dec. 15, 2009**

(54) **HIGH PERFORMANCE MICROPROCESSOR HAVING VARIABLE SPEED SYSTEM CLOCK**

(75) Inventors: **Charles H. Moore**, Woodside, CA (US); **Russell H. Fish, III**, Mt. View, CA (US)

(73) Assignee: **Patriot Scientific Corporation**, San Diego, CA (US)

Reexamination Request:
No. 90/008,306, Oct. 19, 2006
No. 90/008,237, Nov. 17, 2006
No. 90/008,474, Jan. 30, 2007

Reexamination Certificate for:
Patent No.: **5,809,336**
Issued: **Sep. 15, 1998**
Appl. No.: **08/484,918**
Filed: **Jun. 7, 1995**

Certificate of Correction issued May 22, 2007.

Related U.S. Application Data

(62) Division of application No. 07/389,334, filed on Aug. 3, 1989, now Pat. No. 5,440,749.

(51) **Int. Cl.**

G06F 7/76	(2006.01)
G06F 7/48	(2006.01)
G06F 12/08	(2006.01)
G06F 7/78	(2006.01)
G06F 9/30	(2006.01)
G06F 9/32	(2006.01)
G06F 15/76	(2006.01)
G06F 15/78	(2006.01)
G06F 7/52	(2006.01)
G06F 9/38	(2006.01)
G06F 7/58	(2006.01)

(52) **U.S. Cl.** **710/25**; 711/E12.02; 712/E9.016; 712/E9.028; 712/E9.046; 712/E9.055; 712/E9.057; 712/E9.058; 712/E9.062; 712/E9.078; 712/E9.08; 712/E9.081

(58) **Field of Classification Search** ..... None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

3,603,934	A	9/1971	Heath, Jr. et al.
3,696,414	A	10/1972	Allen et al.
3,849,765	A	11/1974	Hamano
3,878,513	A	4/1975	Werner
3,919,695	A	11/1975	Gooding
3,924,245	A	12/1975	Eaton et al.

(Continued)

FOREIGN PATENT DOCUMENTS

EP	0 200 797	A1	11/1986
EP	0 208 287		1/1987
EP	113 516	B1	6/1988

(Continued)

OTHER PUBLICATIONS

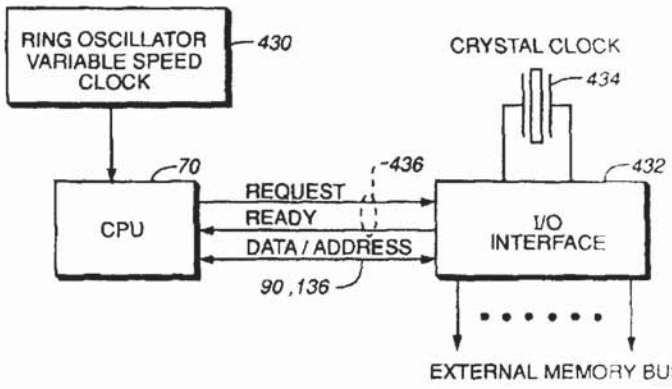
Memorandum Opinion and Order filed Jun. 15, 2007 in 2:05–CV–494 (TJW).
Plaintiffs Technology Property Limited's and Patriot Scientific Corporation's Claim Construction Brief filed Mar. 19, 2007 in 2:05–CV–494 (TJW).

(Continued)

Primary Examiner—Sam Rimell

(57) **ABSTRACT**

A high performance, low cost microprocessor system having a variable speed system clock is disclosed herein. The microprocessor system includes an integrated circuit having a central processing unit and a ring oscillator variable speed system clock for clocking the microprocessor. The central processing unit and ring oscillator variable speed system clock each include a plurality of electronic devices of like type, which allows the central processing unit to operate at a variable processing frequency dependent upon a variable speed of the ring oscillator variable speed system clock. The microprocessor system may also include an input/output interface connected to exchange coupling control signals, address and data with the central processing unit. The input/output interface is independently clocked by a second clock connected thereto.



U.S. PATENT DOCUMENTS

3,967,104	A	6/1976	Brantingham et al.
3,968,501	A	7/1976	Gilbert
3,976,977	A	8/1976	Porter et al.
3,980,993	A	9/1976	Bredart et al.
4,003,028	A	1/1977	Bennett et al.
4,003,033	A	1/1977	O'Keefe et al.
4,037,090	A	7/1977	Raymond, Jr.
4,042,972	A	8/1977	Gruner et al.
4,050,058	A	9/1977	Garlic
4,050,096	A	9/1977	Bennett et al.
4,067,058	A	1/1978	Brandstaetter et al.
4,067,059	A	1/1978	Derchak
4,075,691	A	2/1978	Davis et al.
4,079,455	A	3/1978	Ozga
4,107,773	A	8/1978	Gilbreath et al.
4,110,822	A	8/1978	Porter et al.
4,112,490	A	9/1978	Pohlman et al.
4,125,871	A	11/1978	Martin
4,128,873	A	12/1978	Lamiaux
4,144,562	A	3/1979	Cooper
4,217,637	A	8/1980	Faulkner et al.
4,223,380	A	9/1980	Antonaccio et al.
4,223,880	A	9/1980	Brems
4,224,676	A	9/1980	Appelt
4,236,152	A	11/1980	Masuzawa et al.
4,242,735	A	12/1980	Sexton
4,253,785	A	3/1981	Bronstein
4,255,785	A	3/1981	Chamberlin
4,292,668	A	9/1981	Miller et al.
4,295,193	A	10/1981	Pomerene
4,305,045	A	12/1981	Metz et al.
4,315,305	A	2/1982	Siemon
4,315,308	A	2/1982	Jackson
4,317,227	A	2/1982	Skerlos
4,320,467	A	3/1982	Glass
4,321,706	A	3/1982	Craft
4,328,557	A	5/1982	Gastinel
4,334,268	A	6/1982	Boney et al.
4,335,447	A	6/1982	Jerrim
4,338,675	A	7/1982	Palmer et al.
4,348,720	A	9/1982	Blahut et al.
4,348,743	A	9/1982	Dozier
4,354,228	A	10/1982	Moore et al.
4,358,728	A	11/1982	Hashimoto
4,361,869	A	11/1982	Johnson et al.
4,364,112	A	12/1982	Onodera et al.
4,376,977	A	3/1983	Bruinshorst
4,382,279	A	5/1983	Ugon
4,390,946	A	6/1983	Lane
4,396,979	A	8/1983	Mor et al.
4,398,263	A	8/1983	Ito
4,398,265	A	8/1983	Puhl et al.
4,402,042	A	8/1983	Guttag
4,403,303	A	9/1983	Howes et al.
4,412,283	A	10/1983	Mor et al.
4,425,628	A	1/1984	Bedard et al.
4,449,201	A	5/1984	Clark
4,450,519	A	5/1984	Guttag et al.
4,462,073	A	7/1984	Grondalski
4,463,421	A	7/1984	Laws
4,467,810	A	8/1984	Vollmann
4,471,426	A	9/1984	McDonough
4,472,789	A	9/1984	Sibley
4,488,217	A	12/1984	Binder et al.
4,494,021	A	1/1985	Bell et al.
4,509,115	A	4/1985	Manton et al.
4,538,239	A	8/1985	Magar
4,539,655	A	9/1985	Trussell et al.
4,541,045	A	9/1985	Kromer, III
4,541,111	A	9/1985	Takashima et al.
4,553,201	A	11/1985	Pollack et al.
4,556,063	A	12/1985	Thompson et al.
4,562,537	A	12/1985	Barnett et al.
4,566,063	A	1/1986	Zolnowsky
4,577,282	A	3/1986	Caudel et al.
4,607,332	A	8/1986	Goldberg
4,616,338	A	10/1986	Helen et al.
4,626,798	A	12/1986	Fried
4,626,985	A	12/1986	Briggs
4,626,988	A	12/1986	George
4,627,082	A	12/1986	Pelgrom et al.
4,630,195	A	12/1986	Hester et al.
4,630,934	A	12/1986	Arber
4,641,246	A	2/1987	Halbert et al.
4,649,471	A	3/1987	Briggs et al.
4,660,155	A	4/1987	Thaden et al.
4,660,180	A	4/1987	Tanimura et al.
4,665,495	A	5/1987	Thaden
4,670,837	A	6/1987	Sheets
4,679,166	A	7/1987	Berger et al.
4,680,698	A	7/1987	Edwards et al.
4,689,581	A	8/1987	Talbot
4,691,124	A	9/1987	Ledzius et al.
4,698,750	A	10/1987	Wilkie et al.
4,701,884	A	10/1987	Aoki et al.
4,704,678	A	11/1987	May
4,708,490	A	11/1987	Arber
4,709,329	A	11/1987	Hecker
4,710,648	A	12/1987	Hanamura et al.
4,713,749	A	12/1987	Magar et al.
4,714,994	A	12/1987	Oklobdzija
4,720,812	A	1/1988	Kao et al.
4,724,517	A	2/1988	May
4,739,475	A	4/1988	Mensch, Jr.
4,750,111	A	6/1988	Crosby, Jr. et al.
4,758,948	A	7/1988	May et al.
4,760,521	A	7/1988	Rehwald et al.
4,761,763	A	8/1988	Hicks
4,763,297	A	8/1988	Uhlenhoff
4,766,567	A	8/1988	Kato
4,772,888	A	9/1988	Kimura
4,777,591	A	10/1988	Chang et al.
4,780,814	A	10/1988	Hayek
4,783,734	A	11/1988	May et al.
4,787,032	A	11/1988	Culley
4,794,526	A	12/1988	May et al.
4,797,850	A	1/1989	Amitai
4,803,621	A	2/1989	Kelly
4,805,091	A	2/1989	Thiel et al.
4,809,169	A	2/1989	Sfarti et al.
4,809,269	A	2/1989	Gulick
4,811,208	A	3/1989	Myers et al.
4,816,989	A	3/1989	Finn et al.
4,816,996	A	3/1989	Hill et al.
4,819,151	A	4/1989	May
4,833,599	A	5/1989	Colwell et al.
4,835,738	A	5/1989	Niehaus et al.
4,837,563	A	6/1989	Mansfield et al.
4,837,682	A	6/1989	Culler
4,853,841	A	8/1989	Richter
4,860,198	A	8/1989	Takenaka
4,868,735	A	9/1989	Moller et al.
4,870,562	A	9/1989	Kimoto et al.
4,872,003	A	10/1989	Yoshida
4,882,710	A	11/1989	Hashimoto et al.
4,890,225	A	12/1989	Ellis, Jr. et al.
4,899,275	A	2/1990	Sachs et al.
4,907,225	A	3/1990	Gulick et al.
4,910,703	A	3/1990	Ikeda et al.
4,912,632	A	3/1990	Gach et al.
4,914,578	A	4/1990	MacGregor et al.

4,926,323	A	5/1990	Baror et al.
4,931,748	A	6/1990	McDermott et al.
4,931,986	A	6/1990	Daniel et al.
4,933,835	A	6/1990	Sachs
4,942,553	A	7/1990	Dalrymple et al.
4,956,811	A	9/1990	Kajigaya et al.
4,959,782	A	9/1990	Tulpule et al.
4,967,326	A	10/1990	May
4,967,352	A	10/1990	Keida et al.
4,967,398	A	10/1990	Jamoua et al.
4,974,157	A	11/1990	Winfield et al.
4,979,102	A	12/1990	Tokuume
4,980,821	A	12/1990	Koopman et al.
4,988,892	A	1/1991	Needle
4,989,113	A	1/1991	Asal
4,989,133	A	1/1991	May et al.
4,989,135	A	1/1991	Miki
4,990,847	A	2/1991	Ishimaru et al.
5,008,816	A	4/1991	Fogg, Jr. et al.
5,013,985	A	5/1991	Itoh et al.
5,021,991	A	6/1991	MacGregor et al.
5,022,395	A	6/1991	Russie
5,023,689	A	6/1991	Sugawara
5,031,092	A	7/1991	Edwards et al.
5,036,300	A	7/1991	Nicolai
5,036,460	A	7/1991	Takahira et al.
5,047,921	A	9/1991	Kinter et al.
5,053,952	A	10/1991	Koopman, Jr. et al.
5,068,781	A	11/1991	Gillett, Jr. et al.
5,070,451	A	12/1991	Moore et al.
5,081,574	A	1/1992	Larsen et al.
5,091,846	A	2/1992	Sachs et al.
5,097,437	A	3/1992	Larson et al.
5,103,499	A	4/1992	Miner et al.
5,109,495	A	4/1992	Fite et al.
5,121,502	A	6/1992	Rau et al.
5,127,091	A	6/1992	Boufarah et al.
5,127,092	A	6/1992	Gupta et al.
5,134,701	A	7/1992	Mueller et al.
5,146,592	A	9/1992	Pfeiffer et al.
5,148,385	A	9/1992	Frazier
5,157,772	A	10/1992	Watanabe
5,179,689	A	1/1993	Leach et al.
5,179,734	A	1/1993	Candy et al.
5,226,147	A	7/1993	Fujishima et al.
5,237,699	A	8/1993	Little et al.
5,239,631	A	8/1993	Boury et al.
5,241,636	A	8/1993	Kohn
5,261,057	A	11/1993	Coyle et al.
5,261,082	A	11/1993	Ito et al.
5,261,109	A	11/1993	Cadambi et al.
5,325,513	A	6/1994	Tanaka et al.
5,339,448	A	8/1994	Tanaka et al.
5,353,417	A	10/1994	Fuoco et al.
5,353,427	A	10/1994	Fujishima et al.
5,379,438	A	1/1995	Bell et al.
5,410,654	A	4/1995	Foster et al.
5,410,682	A	4/1995	Sites et al.
5,414,862	A	5/1995	Suzuki et al.
5,421,000	A	5/1995	Fortino et al.
5,440,749	A	8/1995	Moore et al.
5,459,846	A	10/1995	Hyatt
5,511,209	A	4/1996	Mensch, Jr.
5,530,890	A	6/1996	Moore et al.
5,537,565	A	7/1996	Hyatt
5,604,915	A	2/1997	Moore et al.
5,659,703	A	8/1997	Moore et al.
5,784,584	A	7/1998	Moore et al.
6,598,148	B1	7/2003	Moore et al.

FOREIGN PATENT DOCUMENTS

EP	0288649	11/1988
JP	58-25710 A	2/1983
JP	61-127228 A	6/1984
JP	61-138356 A	6/1986
JP	62-145413	6/1987
JP	05-189383	7/1998

OTHER PUBLICATIONS

Defendants' Brief Regarding Construction of Disputed Claim Terms of the 336 and 148 Patents filed Apr. 2, 2007 in 2:05–CV–494 (TJW).

Defendants' Brief Regarding Construction of Disputed Claim Terms of the 584 Patent filed Apr. 2, 2007 in 2:05–CV–494 (TJW).

Plaintiffs' Claim Construction Reply Brief filed Apr. 9, 2007 in 2:05–CV–494 (TJW).

Defendants' Unopposed Motion for Leave to File a Sur–Reply Brief Regarding Claim Construction filed Apr. 19, 2007 in 2:05–CV–494 (TJW).

Defendants' Sur–Reply Brief Regarding Construction of Disputed Claim Terms of the 336 Patent filed Apr. 29, 2007 in 2:05–CV–494 (TJW).

Declaration of Roger L. Cook in Support of Plaintiffs Technology Property Limited's and Patriot Scientific Corporations's Claim Construction Brief filed Mar. 19, 2007 in 2:05–CV–00494 (TJW).

Supplemental Declaration of Roger L. Cook in Support of Plaintiffs Technology Property Limited's and Patriot Scientific Corporations's Claim Construction Brief filed Apr. 9, 2007 in 2:05–CV–00494 (TJW).

Declaration of David J. Lender filed Apr. 2, 2007 in 2:05–CV–00494 (TJW).

Supplemental Declaration of Alvin M. Despain in Support of Plaintiffs' Reply Claim Construction Brief filed Apr. 9, 2007 in 2:05–CV–0049 (TJW).

Declaration of Alvin M. Despain in Support of Plaintiffs' Claim Construction Brief filed Mar. 19, 2007 in 2:05–CV–00494 (TJW).

Nakamura et al., "Microprocessors—Special Purpose," 1987 IEEE International Solid–State Circuits Conference, Feb. 26, 1987.

Transputer Reference Manual, INMOS Limited 1988.

Horwitz et al., "A 20–MIPS Peak, 32–bit Microprocessor with On–Chip Cache," *IEEE Journal of Solid State Circuits*, SC–22(5):790–799 (Oct. 1987).

Submicron Systems Architecture Project, Caltech Computer Science Technical Report, Nov. 1, 1991.

Stevens, C. W., "The Transputer," *IEEE*, pp. 292–300 (1985).

Bosshart et al., "A 533K–Transistor LISP Processor Chip", *IEEE Journal of Solid State Circuits*, SC–22(5): 808–819 (Oct. 1987).

Jguppi et al., "A 20 Mips Sustained 32b CMOS with 64b Data Bus," *IEEE Int'l Solid State Circuits Conf.*, pp. 84–86 (1989).

May, D., "The Influence of VLSI Technology on Computer Architecture," pp. 247–256.

"Motorola MC68HC11A8 HCMOS Single–Chip Microcomputer," table of contents and introduction (1985).

"Motorola MC146805H2, advance information," pp. 1–12.

"MC68332 32–Bit Microcontroller System Integration User's Manual Preliminary Edition, Revision 0.8," (1989).

The Ring Oscillator VCO Schematic.

"INMOS T800 Transputer Data Sheet," (Apr. 1987).
"INMOS T414 Transputer Preliminary Data Sheet," (Feb. 1987).
"INMOS T212 Transputer Preliminary Data Sheet," (Aug. 1987).
"INMOS M212 Disk Processor Product Overview," (Oct. 1987).
Budinich et al., eds. *International Conference on the Impact of Digital Microelectronics & Microprocessors on Particle Physics*, pp. 204–213 (1988).
INMOS Presentation given at Jun. 15, 1988 Workshop on Computer Architecture.
Moore, P., "INMOS Technical Note 15: IMS B005 Design of a Disk Controller board with drives," Dec. 3, 1986.
Matthys R. J., *Crystal Oscillator Circuits*, pp. 25–64 (1983).
Elliot et al., eds. *Scientific Applications of Multiprocessors* Prentice Hall (1988).
Transputer Reference Manual, Cover page, Introduction and pp. 73and 96, INMOS Limited (1988).
"INMOS IMS T414 Transputer," (Jun. 1987).
"INMOS IMS T414 Engineering Data," pp. 107–163.
"INMOS Engineering Data, IMS T414M Transputer, Extended Temperature," (Aug. 1987).
"SM550 High Speed CMOS 4–bit Microcomputer SM–550 Series," (1982) document in Japanese.
"MC88100 RISC Microprocessor User's Manual," Motorola (1989).
Mead et al., eds., *Introduction of VLSI Systems*, Addison Wesley Publishers, (1980).
Moelands, A. P. M., "Serial I/O with the MA B8400 series microcomputers," *Electronic Components and Applications*, 3(1):38–46 (1980).
Stanley, R. C., "Microprocessors in brief," *IBM J. Res. Develop.*, 29(2):110–118 (Mar. 1985).
"MC68332 User's Manual," Motorola (1995).
"TMS370 Microcontroller Family User's Guide," Texas Instruments (1996).
"INMOS Preliminary Data IMS T800 transputer," (Apr. 1987).
"INMOS Engineering Data IMS T212 transputer Preliminary," (Aug. 1987).
"INMOS Product Overview IMS M212 disc processor," (Oct. 1987).
"MN18882 LSI User's Manual," (document in Japanese).
"MN1880 (MN18882) Instruction Manual," (document in Japanese).
"MN188166 User's Manual," (document in Japanese).
Paker, Y., *Multi–Processor Systems*, pp. 1–23 (1983).
"HP Sacajawea External Reference Specification Preliminary Version 1.1," (Jan. 14, 1987).
"Data sheet MOS Integrated Circuit uPD75008," NEC (1989).
Product Brochure by Motorola for MC146805H2.
Shyam, M., "Hardware External Reference Specification for Enhanced Champion/Paladin," Revision of Nov. 11, 1986.
Fish deposition transcript, vols. 1 and 2, held Jun. 25, 2007 and Jun. 26, 2007 in case No. 2–05CV–494 (TJW).
Exhibit 4 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); memo of Sep. 12, 1992 Fish to Higgins re: ShBoom Patents.
Exhibit 5 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); letter of Sep. 11, 1992 Higgins to Falk re: patent application for High Performance Low Cost Microprocessor.
Exhibit 6 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); letter of Sep. 30, 1992 Higgins to Falk re: patent application for High Performance Low Cost Microprocessor.
Exhibit 8 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Mostek 1981 3870/F8 Microcomputer Data Book (1981).
Exhibit 9 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); "IC Master 1980", pp. 2016–2040, published by Fairchild (1980).
Exhibit 10 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); letter of May 12, 1992, Fish to Higgens.
Exhibit 12 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Agreement executed Jan. 3, 1989 between PTA Inc. and Chuck Moore, dba Computer Cowboys.
Exhibit 13 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Oki japan MSH–Boom 96000 Schematic (Jul. 13, 1989).
Exhibit 14 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Assignment of U.S. Appl. No. 07/389,334 from Fish to Fish Family Trust.
Exhibit 15 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Stock Purchase and Technology Transfer Agreement between Fish Family Trust, Helmut Falk, and Nanotronics Corporation (Aug. 16, 1991).
Exhibit 16 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); fax of Jul. 16, 2004Mmarshall to Suanders and Heptig w/ attached Jul. 15, 2004 memo from Beatie re: Fish and Moore.
Exhibit 17 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Fax of Jul. 29, 2004, Heptig to Marshall with attached executed agreement.
Exhibit 18 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Plaintiffs' Second Amended Complaint filed Sep. 22, 2006 in 3:06–CV–00815.
Exhibit 19 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Settlement Agreement between Patriot Scientific Corporation, Fish, and the trustee of Fish Family Trust.
Exhibit 21 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Comparisons of RISC Chips (Dec. 11, 1988).
Exhibit 22 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); VL86C010 An Affordable 32–bit RISC Microprocessor System, VLSI Technology, Inc.
Exhibit 23 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); VY86Cxxx ARM 32–Bit CMOS product literature, EDN (Nov. 21, 1991).
Exhibit 24 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); letter of Apr. 26, 1989 Ireland to Fish w/ copy of Apr. 17, 1989 article in Electronic News titled "35ns 256K Device, VLSI Debuts SRAM Designed With Hitachi,".
Exhibit 28 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Introduction to RISC Technology, LSI Logic Corporation (Apr. 1988).
Exhibit 29 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); "SH–Boom Patent Documentation," (Jun. 21, 1989).
Exhibit 30 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); "Sh–Boom Licensing Strategy," (Jan. 19, 1990).

Exhibit 31 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); "Transputer Includes Multiprocessing protocol," Jan. 2, 1991.
Exhibit 32 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); article titled "INMOS details next Transputer," (Apr. 18, 1991).
Exhibit 33 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Articles from Electronic World News.
Exhibit 35 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); INMOS Preliminary Data IMS T414 transputer.
Exhibit 36 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Comparison of Intel 80960 and Sh–Boom Microprocessors (1989).
Exhibit 37 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Memo of Jul. 13, 1989, Fish to Chu w/ attached comparison of MIPS 2000 to Sh–Boom.
Exhibit 38 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); fax of Jun. 10, 1992 Fish to Higgins w/ attached document titled State of the Prior Art ShBoom Microprocessor.
Exhibit 39 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Apr. 12, 1989 Time and Responsibility Schedule.
Exhibit 40 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); handwritten note.
Exhibit 41 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); memo of Jun. 1989 to PT Acquisitions, Inc. re: fees due for searches conducted.
Exhibit 42 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); memo of Jun. 28, 1992 Fish to Higgins re: Dialog Patents re: ShBoom.
Exhibit 43 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); letter of Aug. 6, 1998 Haerr to Turner transmitting documents.
Exhibit 44 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); Declaration of Moore re: U.S. Appl. No. 08/484,918.
Exhibit 45 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); fax of Nov. 3, 1989, Leckrone to Fish with attached draft license agreement between PT Acquisitions and Oki Electric Industries.
Exhibit 46 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); letter of Nov. 29, 1989 Fish to Slater re: Japanese "borrowing" Sh–Boom 50 MHz RISC Chip.
Exhibit 47 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); letter of Dec. 29, 1989 Leckrone to Fish re: ShBoom project.
Exhibit 48 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); letter of Jul. 16, 1990 Fish to Leckron re: attorney client relationship and conflict of interest.
Exhibit 49 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); letter of Jul. 24, 1990 Leckrone to Fish re: letter of Jul. 16, 1990 (EX 48).
Exhibit 50 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); letter of Aug. 27, 1990 Moore to Fish re: ShBoom confidentiality.
Exhibit51 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); PT Acquisitions / Alliance Semiconductor Corp. Manufacturing Agreement (Jul. 20 1990).
Exhibit 52 to Fish deposition of Jun. 25–26, 2007 in case No. 2–05CV–494 (TJW); letter of Feb. 6, 1990 to PT Acquisitions from Dun & Bradstreet Receivable Recovery Systems re: final notice for payment of account.
National Semiconductor HPC16400/HPC36400/HPC46400 High–Performance microControllers with HDLC Controller product literature.
NEC Data Sheet Mos Integrated Circuit uPD7225 Programmable LCD Controller/Driver. Part Nos. uPD7225G00, uPD7225G01, uPD7225GB–3B7.
NEC Electronics Inc. High–End, 8–Bit, Single–Chip CMOS Microcomputers product literature.
NEC Electronics Inc. uPD78C10/C11/C14 8–Bit, Single–Chip CMOS Microcomputers with A/D Converter product literature.
NEC Electronics Inc. Microcomputer Products Single–Chip Products Data Book vol. 1 of 2 cover page.
NEC Electronics Inc. Microcomputer Products Microprocessors, Peripherals, & DSP Products Data Book vol. 2 of 2 cover page.
NEC Electronics Inc. MOS Integrated Circuit uPD70208H, 70216H Data Sheet, V40HL, V50HL, 16/8; 16–Bit Microprocessor.
NEC Electronics Inc. MOS Integrated Circuit uPD7225 Programmable LCD Controller/Driver.
Signetics Microprocessor Data manual cover page.
Signetics Microprocessor Products Data manual, 8×330 Floppy Disk Formatter/Controller product specification.
Signetics Microprocessor Products Data manual, SC96AH Series Single–Chip 16–Bit Microcontrollers preliminary specification.
Realtime DSP; The TMS320C30 Course, Revision 3 (educational document re programming Digital Signal Processor microprocessor).
Texas Instruments TMS320C30 Digital Signal Processor product literature.
Texas Instruments TMS320C30 Digital Signal Processor product literature.
Texas Instruments TMS34010 Graphics System Processor product literature.
Texas Instruments TMS320 DSP Designer's Notebok, Using a TMS320C30 Serial Port as an Asynchronous RS–232 Port Application Brief: SPRA240.
UK application 8233733 as filed Nov. 26, 1982.
Office Action of Jan. 31, 2000 in U.S. Appl. No. 09/124,623.
TPL Infringement Contention for the TLCS–900/L Series.
TPL Infringement Contention for Toshiba Microcontroller.
TPL Infringement Contention for Toshiba TC35273.
Duell, C. H., "Everything that can be invented has been invented," 2 pages downloaded from http://www.tplgroup.net/patents/index.php.
Alliacense Product Report, NEC Microcontroller, UPD789473, B Bit Microcontroller, pp. 1–38.
Mostek Corp., Advertisement, EDN, Nov. 20, 1976.
Guttag, K.M., "The TMS34010: An Embedded Microprocessor," *IEEE Micro*, 8(3):39–52 (May 1988).
"8–Bit Single–Chip Microprocessor Data Book," Hitachi America Ltd., Table of Contents and pp. 251–279) Jul. 1985).
Request for Reexamination of U.S. Patent 6,598,148 as filed Sep. 21, 2006.
Office action of Nov. 22, 2006 in application No. 90/008,227.
Toshiba TLCS–42, 47, 470 User's Manual Published in Apr. 1986.
Fukui et al., "High Speed CMOS 4–bit Microcomputer SM550 Series," published 1982, 1983.

Intel 80386 Programmer's Reference Manual, published by Intel (1986).
Fairchild Microcomputers, F8/3870, F6800, Bit Slice, IC Master 1980, pp. 1, 2016–2040 (1980).
Mostek Corp., "Mostek 1981 3870/F8 Microcomputer Data Book", Feb. 1981, pp. III–76 through III–77, III–100 through III–129, and VI–1 through VI–31.
IBM Systems Reference Library, IBM System/360 Model 67 Functional Characteristics, File No. S360–01, Form A27–2719–0, published by IBM (1967).
Anderson, D.W., The IBM System/360 Model 91: Machine Philosophy and Instruction Handling, IBM Journal, (1967).
GE–625/ 635 Programming Reference Manual, revised Jan. 1966.
Clipper™ 32–Bit Microprocessor, Introductions to the Cliper Architecture, published by Fairchild in 1986.
Simpson et al., "The IBM RT PC Romp processor and memory management unit architecture," *IBM systems Journal*, 26(4):346–360.
M68300 Family MC68332 User's Manual, published by Motorola, Inc. in 1995.
Ditzel et al., "The Hardware Architecture of the Crisp Microprocessor," AT & T Information Systems, ACM, pp. 309–319 and table of contents (1987).
i860 64–Bit Microprocessor, published by Intel Corporation Feb. 1989.
Rau et al., "The Cydra 5 Departmental Supercomputer, Design Philosophies, Decisions, and Trade–offs," *IEEE* pp. 12–36 (1989).
Datasheet for Intel 4004 Single Chip 4–Bit 9–Channel Microprocessor, pp. 8–15 to 8–23.
Intel MCS–4 Micro Computer Set (Nov. 1971).
Intel 8008 8–Bit Parallel Central Processor Unit published by Intel (Nov. 1972).
iAPX 386 High Performance 32–Bit Microprocessor Product Review, published by Intel (Apr. 1984).
Intel 80386 Programmer's Reference Manual, published in Intel (1986).
Motorola MC68020 32–Bit Microprocessor User's Manual (1984).
Thornton, J. E., "Design of a Computer, The Control Data 6600," published by Advanced Design Laboratory (1970).
6400/6500/6600 Computer Reference Manual, published by Control Data® (1965, 1966, 1967).
Grishman, R., "Assembly Language Programming for the Control Data 6600 and Cyber Series Algorithmics".
Hennessy et al., "MIPS: A Microprocessor Architecture," *IEEE*, pp. 17–22 (1982).
Hennessy et al., "Hardware/software tradeoff for increased performance," *Proceedings of the Symposium on Architectural Support for Programming Languages and Operating Systems*, pp. 2–11, ACM, Apr. 1982.
Hennessy et al., "MIPS: A VLSI Processor Architecture, Technical Report 223," Computer Systems Laboratory, Department of Electrical Engineering and Computer Science, Standford University Nov. 1981.
Gross et al., "Measurement and evaluation of MIPS architecture and processor," *ACM Trans. Computer Systems*, pp. 229–257 Aug. 1988.
Bit Sparc Integer Unit B5000 Datasheet attached to a presentation by Anant Agrawal in 1989.
Sequin et al., "Design and Implementation of RISC1," pp. 276–298 from *VLSI Architecture*, B. Randell and P.C. Treleaven, editors, Prentice Hall, 1983.
Ungar et al., "Architecture of SOAR: Smalltalk on a RISC," *Proceedings of the 11th Annual International Symposium on Computer Architecture ISCA '84*. ACM Press, New York, NY, pp. 188–197 (1984).
Cray–I Computer System Hardware Reference Manual, Publication No. 2240004, Rev C, Nov. 4, 1997.
Acorn Computers, Ltd., Acorn RISC Machine CPU Software Manual, Issue 1.00 Oct. 1985.
Patterson et al., "Architecture of a VLSI Instruction Cache for A RISC," *ACM*, pp. 108–116 (1983).
Patterson, D. A., "Reduced Instruction Set Computers" *Communication of the ACM*, 28(1):8–21, Jan. 1985.
Patterson, D. A., "RISC watch", pp. 11–19 (Mar. 1984).
Sherburne, R. W., "Processor Design Tradeoffs in VLSI," U.C. Berkeley, May 1984. PhD Dissertation.
Excerpt from A Seymour Cray Perspective http://research.Microsoft.com/users/gbell/craytalk/std001.htm (Slide 1).
Excerpt from A Seymour Cray Perspective htpp.//research.microsoft.com/users/gbell/craytalk/std029.htm (Slide 29).
RISC Roots: CDC 6600 (1965).
http://www.bitsavers.org/pdf/tdc/6x00/.
Simpson, R.O., "The IBM RT Personal computer," *BYTE*, 11(11):43–78 (Oct. 1986).
Ryan, D.P., "Intel's 80960: An Architecture Optimized for Embedded Control," *IEEE, Micro,* published in Jun. 1988.
Waters, F., "IBM RT Personal Computer Technology," IBM Corp. 1986.
Alliacense Product Report, USP 5784584, TLCS–900/H1 Series TMP92C820, Toshiba Microcontroller TMP92C820 / TLCS–900/H1 Series 16–bit Microcontroller, pp. 1–9, filed Aug. 14, 2006 in 2:05–CV–00494–TJW.
Alliacense Product Report, NEC Microcomputer, USP 5784584, V850E2 32 Bit Microcontroller, pp. 1–8 (2006).
"8 bit Dual 1–chip Microcomputer MN1890 Series User's Manual," translation of original Japanese language document, by Matsushita Electric Industrial Co., Ltd. Semiconductor Sales Division.
Sibigtroth, J. M., "Motorola's MC68HC11: Definition and Design of a VLSI Microprocessor," *IEEE Micro* 4(1):54–65 (1984).
"Specification Sheet, MN18882 (Book)," translation of the Japanese language original, Code No. MIG0175, Matsushita Electronics Industry Corporation, Microcomputer Products Division, Oct. 2, 1990.
"DS5000 Soft Microcontroller User's Guide Preliminary V 1.0," Dallas Semiconductor.
MN188166 User's Manual, Japanese language document with English translation.
Alliacense Product Report—Preliminary Review, USP 5,440,749; GPS Car Navigation Main Processor.
Alliacense Product Report—Preliminary Review, USP 5,440,749; GSP Navigation System GPS Chipset.
Alliacense Product Report—Preliminary Review, USP 5,440,749; GPS Navigation System Main Microprocessor.
Alliacense Product Report—Preliminary Review, USP 5,440,749; Kyocera Digital Camera Finecam S3R Image Processor.
Alliacense Product Report—Preliminary Review, USP 5,440,749; PDA/Mobile Navigation, GPS Processor.
Motorola MC68020 32–bit Microprocessor User's Manual, 2nd Edition, Rev. 1, Prentice–Hall, 1985.
Barron et al., "The Transputer," *Electronics*, pp. 109–115 (1983).

Burroughs Corporation, "Burroughs B7700 Systems Reference Manual," 1973.
Fiasconaro, J., "Microarchitecture of the HP9000 Series 500 CPU," *Microarchitecture of VLSI Computers, NATO ASI Series No. 96*, Antognetti, eds., pp. 55–81.
MacGregor et al., "The Motorola MC68020," *IEEE Micro*, 4(4):103–118 (1984).
Best et al., "An Advanced–Architecture CMOS/SOS Microprocessor", IEEE Micro, vol. 2, No. 3, vol. 2, No. 3 (Jul. 1982), pp. 10–26.
Technology Properties Limited (TPL), Moore Microprocessor Patent (MMP) Portfolio, downloaded from <<www.tpl-group.net/patents/index.php>> downloaded on Aug. 3, 2006, 3 pages total.
Acorn's RISC Leapfrog, Acorn User special issue, Jun. 1987; 59: 149–153.
Agrawal, et al., "Design Considerations for a Bipolar Implementation of SPARC," Compcon Spring apos;88. Thirty–Third IEEE Computer Society International Conference, Digest of Papers, Feb. 29–Mar. 3, 1988, pp. 6–9.
Agrawal, "An 80 MHz Bipolar ECL Implementation of SPARC," Sun Microsystems, Inc., Jun. 25, 1989, 40 pages total.
ARM Datasheet, Part No. 1 85250 0360 0, Issue No. 1 (Mar. 17, 1987).
Atmel SPARC Instruction Set, "Assembly Language Syntax," Rev. 4168C–AERO–08/01, 2002.
Bagula, "A 5V Self–Adaptive Microcomputer with 16Kb of #2 Program Storage and Security," IEEE International Solid–State Circuit Conference, 1983, pp. 34–35.
Bayko, Great Microprocessors of the Past and Present (V 11.7.0), downloaded from: <<http://web.archive.org/web/20010107210400/http://bwrc.eecs.berkeley.edu/CIC/Archive/cup_history.html>>, Feb. 2007, 60 pages total.
Books Review: Operating Systems A Systematic View, William S. Davis, Addison–Wesley Publishing Company, Inc., 1987; 26(4):453–454.
Bourke, "Character Synchronization During Overrun Conditions," Delphion, IBM Technical Disclosure Bulletin, Dec. 1977.
Burroughs Corporation, "Burroughs B5500 Information Processing System Reference Manual," 1973.
CAL Run Fortran Guide, University of California, Computer Center, Berkeley, 292 pages total, (Sep. 1974).
CDC 6000 Computer Systems—Cobol Instant 6000, Version 3; Control Data Publication No. 60327600A (Apr. 1971).
CDC 6000 Computer Systems, 7600 Computer Systems: Fortran Extended Instant 6000 Version 3, 7600 Version 1; Control Publication No. 60305900A (May 1971).
CDC 6000 Computer Systems/7600 Computer Systems: Fortran Extended Reference Manual, 6000 Version 3, 7600 Version 1; Control Data Publication No. 60329100, Revision D (1972).
CDC 6000 Series Computer System, 6642 Distributive Data Path Operation and Programming Reference Manual; Control Data Publication No. 60376300C (Mar. 1979).
CDC 6000 Series Computer Systems—Chippewa Operating System Documentation, vol. I, Preliminary Edition (updated May 1966).
CDC 6000 Series Computer Systems—Chippewa Operating System Documentation, vol. II, Preliminary Edition, Peripheral Packages and Overlays (Oct. 1965).
CDC 6000 Series Computer Systems—Chippewa Operating System Documentation, vol. III, Preliminary Edition, DSD—The Systems Display, (Nov. 1965).
CDC 6000 Series Computer Systems Ascent General Information Manual; Control Data Publication No. 60135400 (Feb. 66).
CDC 6000 Series Computer Systems Chippewa Operating System Reference Manual; Control Data Publication No. 60134400 (Dec. 1965).
CDC 6000 Series Computer Systems Hardware Reference Manual; Control Data Publication No. 60100000 (Aug. 1978).
CDC 6000 Series Computer Systems/ 7600 Computer System: 6000 Compass Version 2, 7600 Versions 1 & 2 Reference Manual; Control Data Publication No. 60279900, Revision D, (1970, 1971, 1972).
CDC 6000 Series Computer Systems: Chippewa Operating System Fortran Reference Manual; Control Data Publication No. 60132700A (May 1966).
CDC 6000 Series Computer Systems: Peripheral Equipment Reference Manual; Control Data Publication No. 60156100, Revision J (Mar. 1970).
CDC 6000 Series Computer Systems: Site Preparation and Installation Manual; Control Data Publication No. 60142400, Revision B (Sep. 1965).
CDC 6000 Series Computer Systems: Fortran Extended General Information, Control Data Publication No. 60176400 (Oct. 1966).
CDC 6000 Series Fortran Extended 4.0, Internal Maintenance Specifications , (1971).
CDC 6000 Series Introduction and Peripheral Processors Training Manual, Second Edition, Control Data Publication No. 60250400 (Nov. 1968).
CDC 6400 Central Processor, Control Data Publication No. 60257200 (Feb. 1967).
CDC 6400/6500/6600 Ascent–to–Compass Translator; Control Data Publication No. 60191000 (Mar. 1967).
CDC 6400/6500/6600 Computer System Input/Output Specification; Control Data Publication No. 60045100 (Sep. 1967).
CDC 6400/6500/6600 Computer System Instant SMM; Control Data Publication No. 60299500 (Nov. 1969).
CDC 6400/6500/6600/ Computer Systems Compass Reference Manual; Data 60190900, Revision B (Mar. 1969).
CDC 6400/6500/6600 Computer Systems Reference Manual; Control Data Publication No. 60100000, Revision K (Aug. 1970).
CDC 6400/6500/6600 Extended Core Storage Systems Reference Manual; Control Data Publication No. 60225100, (Feb. 1968).
CDC 6400/6500/6600 Jovial General Information Manual; Control Data Publication No. 60252100A (Mar. 1969).
CDC 6400/6500/6600 Computer Systems: Ascent/Asper Reference Manual; Control Data Publication No. 60172700 (Jul. 1966).
CDC 6400/6600 Fortan Conversion Guide; Data Publication No. 60175500 (Aug. 1966).
CDC 6400/6600 Systems Bulletin (Oct. 10, 1968), 84 pages.
CDC 6400/6600, Export/Import 8231 Reference Manual; Data Publication No. 60189100 (Apr. 1967).
CDC 6600 Central Processor vol. 1; Control & Memory; Data Control Publication Nol. 021067 (Mar. 1967).
CDC 6600 Central Processor, vol. 2, Functional Units; Control Data Publication No. 60239700 (Mar. 1967).

CDC 6600 Chassis Tabs; Control Data Publication No. 63016700A (Apr. 1965).
CDC 6600 Chassis Tabs; Control Data Publication No. 63019800 (Mar. 1965).
CDC 6600 Chippewa Operating System; Control Data Publication No. 60124500, Revision C (Apr. 1965).
CDC 6600 Computer System 6601 A–J, 6613A/B/C, 6604A/B/C, 6614–A/B/C/ Central Processor (including Functional Units) vol. 1, Diagrams & Circuit Description; Control Data Publication No. 60119300BT (Jan. 1968).
CDC 6600 Computer System 6601 A–J, 6613A/B/C, 6604A/B/C/, 6614–A/B/C Peripheral and Control Processor/Central Memory/Clock/Extended Core Storage Coupler (Std Opt 10102 and/or Spec Opt 60080)/Power Wiring, vol. 2, Diagrams & Circuit Descriptions; Control Data Publication No. 60119300BT (Jan. 1968).
CDC 6600 Computer System Operating System/Reference Manual, SIPROS66, 1st ed.; Control Data Publication No. 60101800A (1965).
CDC 6600 Computer System Programming System/Reference Manual, vol. 1. Ascent; Control Data Publication No. 601016008 (1965).
CDC 6600 Computer System Programming System/Reference Manual, vol. 2, Asper; Control Data Publication No. 60107008 (1965).
CDC 6600 Computer System Programming vol. 3, Fortran 66; Control Data Publication No. 60101500B (1965).
CDC 6600 Computer Training Manual vol. 2, Preliminary Edition, Section 7.2 Shift Functional Units, 164 pages.
CDC 6600 Data Channel Equipment 6602–B/6612–A, 6603–B, 6622–A, 6681–B, 6682–A6683–A, S.O. 60022, 60028, 60029, Diagrams & Circuit Description' Control Data Publication No. 60125000, Revision G (Jun. 1966).
CDC 6600 Training Manual; Control Data Publication No. 60147400, Revision A (Jun. 1965).
CDC 6603—A/B/C Disk File Controller Reference Manual; Control Data Publication No. 60334000 (1970).
CDC 6638 Disk File Subsystem Training Supplement; Control Data Publication 602500800A (Oct. 1968).
CDC 6638 Disk File System Standard Option 10037–A, 6639–A/B File Controller—Diagrams and Circuit Description/Maintenance/Maintenance Aids/Parts List/Wire Lists/Chassis Tabs; Control Data No. 60227300, Revision H (Mar. 1974).
CDC 6639—A/B Disk File Controller Reference Manual; Control Data Publication No. 60334100E (Aug. 1973).
CDC 6639 Disk Controller Training Manual Test Edition (Sep. 1967), 28 pages.
CDC APL Version 2 Reference Manual, CDC Operating Systems :NOS; Control Data Publication No. 60454000F (Nov. 1980).
CDC Basic Version 3 Reference Manual: NOS 1 & NOS/BE 1; Control Data Publication No. 19983900G (Oct. 1980).
CDC Chippewa Operating System, Document Class ERS, System No. E012, Version 1.1; External Reference Specifications—Chippewa Operating System, (Jun. 1966).
CDC Chippewa Operating System, Document Class SIR, System No. E012, Version 1.0, (Mar. 3, 1966).
CDC Chippewa Operating System, Document Class SIR, System No. E012, Version 1.1, (Mar. 3, 1966).
CDC Cobol Version 4 Instant Manual, Cyber 170 Series, Cyber 70 Models 72, 73, 74, 6000 Series; Control Data Publication No. 60497000A (Feb. 1976).
CDC Cobol Version 5 Instant Manual, Operating Systems: NOS 1/BE 1; Control Data Publication No. 60497300B (Feb. 198).
CDC Codes/Control Data 6400/6500/6600 Computer Systems; Control Data Publication No. 60141900, Revision C (1966, 1967).
CDC Codes/Control Data 6400/6500/6600/6700 Computer Systems; Control Data Publication No. 60141900, Revision D (1966, 1967).
CDC Codes/Control Data 6600; Control Data Publication No. 60141900, Revision A (May 1965).
CDC Compass Version3 Instant, Operating Systems: NOS 1, NOS 2, NOS/BE 1, Scope 2; Control Data Publication No. 60492800D (Jun. 1982).
CDC Course No. FH4010–1C, NOS Analysis, Student Handbook, Revision C (Apr. 1980).
CDC Course No. FH4010–4C NOS Analysis, Study Dump (Apr. 1980).
CDC Cyber 170 Models 720, 730, 740, 750 and 760 Model 176 (Level B/C) Computer Systems, Codes; Control Data Publication No. 60456920C, (Jun. 1981).
CDC Cyber 70 Computer Systems Models 72, 73, 74, 6000 Computer Systems: Fortran Reference Manual Models 72, 73, 74 Version 2.3, 6000 Version 2.3; Control Data Publication No. 19980400, Revision F (Jul. 1972).
CDC Cyber 70 Computer Systems Models 72, 73, 74, 76, 7600 Computer System, 6000 Computer Systems—ALGOL Reference Manual, Cyber 70 Series Version 4, 6000 Series Version 4, 7600 Series Version 4; Control Data Publication No. 60384700A (Aug. 1973).
CDC Cyber 70 Computer Systems Models 72, 73, 74, 76, 7600 Computer System, 6000 Computer Systems: Cobol Instant Models 72, 73, 74 Version 4, Model 76 Version 1, 6000 Version 4; Control Data Publication No. 60328400A (Dec. 1971).
CDC Cyber 70 Computer Systems Models 72, 73, 74, 76, 7600 Computer System, 6000 Computer Systems: Fortran Extended Instant Models 72, 73, 74 Version 4, Model 76 Version 2, 7600 Version 2, 6000 Version 4; Control Data Publication No. 60357900A (Nov. 1971).
CDC Cyber 70 Computer Systems Models 72, 73, 74, 76 7600 Computer System, 6000 Computer Systems: Fortran Extended Reference Manual Models 72, 73, 74 Version 4, Model 76 Version 2, 6000 Version 4; Control Data Publication No. 60306600A (Oct. 1971).
CDC Cyber 70 Series 6000 Series Computer Systems: APL *Cyber Reference Manual; Control Data Publication No. 19980400B (Jul. 1973).
CDC Cyber 70 Series Computer Systems 72, 73, 74, 6000 Series Computer Systems—Kornos 2.1 Workshop Reference Manual; Control Data Publication No. 97404700D (1976).
CDC Cyber 70 Series Models 72/73/74, 6000 Series Computer Systems, Krono 2.1 Operator Guide; Control Data Guide; Control Data Publication 60407700A (Jun. 1973).
CDC Cyber 70 Series Models 72/73/74, 6000 Series Computer Systems, Kronos 2.1 Installation Handbook; Control Data Publication No. 60407500A (Jun. 1973).
CDC Cyber 70 Series Models 72/73/74, 6000 Series Computer Systems, Kronos 2.1 Time–Sharing User's Reference Manual; Control Data Publication No. 60407600B (May 1974).

CDC Cyber 70/ Model 76 Computer System, 7600 Computer System: Fortran Run, Version 2 Reference Manual; Control Data 60360700C (May 1974).
CDC Cyber Interactive Debug Version 1 Guide for Users of Fortran Extended Version 4, CDC Operating Systems: NOS 1, NOS/ BE1, Control Data Publication No. 60482700A (Feb. 1979).
CDC Cyber Interactive Debug Version 1 Guide for Users of Fortran Version 5, Operating Systems: NOS 1, NOS/ BE 1; Control Data Publication No. 6048100C (Sep. 1984).
CDC Cyber Interactive Debug Version 1 Reference Manual, NOS 1, NOS 2, NOS/ BE 1; Data Control Publication No. 60481400D (Jun. 1984).
CDC Cyber Loader Version 1 Instant, CDC Operating System NOS1, NOS/ BE 1; Control Data Publication No. 60449800C (Aug. 1979).
CDC Disk Storage Subsystem—Operation and Programming Manual; Control Data Publication No. 60383900, Version T (1972–1980).
CDC Fortran Extended 2.0, Document Class ERS, System No. C012, (Dec. 1966).
CDC Fortran Extended 2.0, Document Class IMS, Internal Maintenance Specifications—64/65/6600 V Fortran Extended Version 2 (Mar. 1969).
CDC Fortran Extended Version 4 Instant Manual, Operating Systems: NOS 1, NOS/BE, 1 Scope 2; Control Data Publication No. 60497900B (Jun. 1981).
CDC Fortran Extended, Sales Technical Memorandum (May 1967).
CDC Fortran Version 5 Instant, CDC Operating System: NOS 1, NOS/ BE 1, Scope 2; Control Data Publication No. 60483900A (Jan. 1981).
CDC GED Fortran Extended 1.0, Product No. C012, Dept. No. 254, Project No. 4P63FTN (Aug. 1967).
CDC Instant 6400/3500/6500 Simula; Control Data Publication No. 60235100, Revision A (Feb. 1969).
CDC Instant 6400/6500/6600 Compass; Control Data Publication No. 60191900, Revision A (1968).
CDC Instant Fortran 2.3 (6000 Series); Data Publication No. 60189500D (May 1969).
CDC Internal Maintenance Specification; Fortran V5.; Control Data Publication No. 77987506A.
CDC Internal Maintenance Specification; Fortran V5.; Control Data Publication No. 77987506A.
CDC Kronos 2.1 Reference Manual vol. 1 of 2; Control Data Cyber 70 Series Models 72/73/74, 6000 Series Computer Systems; Control Data Publication No. 60407000D (Jun. 1975).
CDC Kronos 2.1 Time–Sharing User's Reference Manual, Cyber 70 Series Models 72, 73, 74, 6000 Series Computer Systems; Control Data Publication No. 60407600D (Jun. 1975).
CDC Model dd 60A Computer Control Console/Customer Engineering Manual; Control Data Publication No. 82100010 (Mar. 1965).
CDC Model dd60b Computer Control Console/Customer Engineering Manual/ Control Data Publication No. 82103500 (Feb. 1967).
CDC Network Products: Network Access Method Version 1, Network Definition Language Reference Manual; Control Data Publication No. 60480000J (May 1981).
CDC Network Products: Network Terminal User's Instant—Operating System NOS 1; Control Data Publication No. 60456270C, (Oct. 1980).
CDC NOS 2 Operations Handbook; Control Data Publication No. 60459310, (Aug. 1994).
CDC NOS Version 1 Applications Programmer's Instant, CDC Computer Systems; Cyber 170 Series, Cyber 70 Models, 71, 72, 73, 74, 6000 Series; Control Data Publication No. 60436000H (Jan. 1980).
CDC NOS Version 1 Internal Maintenance Specification vol. 1 of 3; Control Data Publication No. 60454300B (Aug. 1979).
CDC NOS Version 1 Internal Maintenance Specification vol. 2 of 3; Control Data Publication No. 60454300B (Aug. 1979).
CDC NOS Version 1 Internal Maintenance Specification vol. 3 of 3; Control Data Publication No. 60454300B (Aug. 1979).
CDC NOS Version 1 Operator's Guide, CDC Computer Systems: Cyber 170 Series, Cyber 70 Model 71, 72, 73, 74, 6000 Series (Dec. 1980).
CDC NOS Version 1 Reference Manual vol. 1 of 2, CDC Computer Systems: Cyber 170 Series, Cyber 70 Models 71,72, 73, 74, 6000 Series; Control Data Publication No. 60435400J (1979).
CDC NOS Version 1 Reference Manual vol. 2 of 2, CDC Computer Systems: Cyber 170 Series, Cyber 70 Models 71, 72, 73, 74, 6000 Series; Control Data Publication No. 60445300E (1977).
CDC NOS Version 1 System Maintenance Reference Manual; Control Data Publication No. 60455380H (Apr. 1981).
CDC NOS Version 2 Analysis Handbook, Cyber 180, Cyber 170, Cyber 70 Models 71, 72, 73, 74, 6000; Control Data Publication No. 60459300D (Oct. 84).
CDC NOS Version 2 Analysis Handbook; Control Data Publication No. 60459300U (Jul. 1994).
CDC NOS Version 2 Installation Handbook, Cyber 180, Cyber 170, Cyber 70 Models 71, 72, 73, 74, 6000; Control Data Publication No. 60459320E_(Mar. 1985).
CDC NOS Version 2 Operation/Analysis Handbook, Cyber 170, Cyber 70 Models 71, 72, 73, 74 6000, Control Data Publication No. 60459310C (Oct. 1983).
CDC NOS Version 2 System Maintenance Reference Manual, Cyber 170, Cyber 70 Models 71, 72, 73 74, 6000; Control Data Publication No. 60459300C (Oct. 1983).
CDC NOS/BE Version 1 Diagnostic Handbook, Cyber 180; Cyber 170, Cyber 70 Models 71, 72, 73, 74, 6000; Control Data Publicaton No. 60494400–V (1986).
CDC NOS/BE Version 1 Installation Handbook, Cyber 180, Cyber 170, Cyber 70 Models 71, 72, 73, 74, 6000; Control Data Publication No. 60494300AB (Dec. 1988).
CDC NOS/BE Version 1 Reference Manual, Cyber 170 Series, Cyber 70 Models 71, 72, 73, 74, 6000 Series; Control Data Publication No. 60493800M 1981.
CDC Outline of Reports on Feasibility Study of 64/6600 Fortran Ver 3.0 and Conversational Fortran, Fortran Study Project, Product No. X010, Dept. No. 254, Project No. 4P63, (Jun. 1966).
CDC Pascal Version 1 Reference Manual, Operating Systems: NOS 2; Control Data Publication No. 60497700 (Sep. 1983).
CDC Pascal Version 1 Reference Manual, Operating Systems: NOS 2; Control Data Publication No. 60497700A (Dec. 1982).

CDC PL/1 Version 1 Instant, CDC Operating Systems: NOS 1, NOS/ BE1, Control Data Publication No. 60483700A (Nov. 1979).
CDC Simscript 11.5 Instant, Control Data Publication No. 84000450B (Sep. 1978).
CDC Sort/Merge Version 4 and 1 Instant, Operating Systems: NOS 1, NOS/BE 1, Scope 2; Control Data Publication No. 60497600C (Jan. 1981).
CDC Sort/Merge Vision 5 Reference Manual, Operating Systems: NOS 2, NOS/ BE 1; Control Data Publication No. 60484800C (Feb. 1984).
CDC SYMPL Version 1 Instant, NOS 1, NOS/BE 1, Scope 2; Control Data Publication No. 60482600A (May 1978).
CDC SYMPL Version 1 Users Guide, Operating Systems: NOS 1, NOS/BE 1, Scope 2; Control Data Publication No. 60499800B (Apr. 1978).
CDC Update Instant, Cyber 170 Series, Cyber 70 Series, 6000 Series, 7600 Computer Systems; Control Data Publication No. 60450000A (Nov. 75).
CDC Update Reference Manual Operating Systems: Scope 3.4, Kronos 2.1; Control Data Publication No. 60342500, Revision H (1971–1976).
CDC Xedit Version 3 Reference Manual, Operating System: NOS 1; Control Data Publication No. 60455730B (Aug. 1979).
Chippewa Laboratories Fortran Compiler Run, Preliminary Edition, CDC 6000 Series Computer Systems, (Apr. 1966).
Cho et al., WAM 3.6: A 40K Cache Memory and Memory Management Unit, ISSCC '86, Feb. 19, 1986.
Cordell, II et al., "Advanced Interactive Executive Program Development Enviornment," IBM Systems Journal, 1987; 26(4):381–382.
Crawford, "The i486 Executing Instructions in One Clock Cycle," IEEE Micro, pp. 28–36 (Feb. 1990).
Disk Routines and Overlays, Chippewa Operating System, CDC Development Division—Applications, (Nov. 1965).
Dowsing et al., "Computer Architecture: A First Course, Chapter 6: Architecture and the Designer," Van Nostrand Reinhold (UK) Co. Ltd., pp. 126–139.
Evans et al., "An Experimental 512–bit Nonvolatile Memory with Ferroelectric Storage Cell," IEEE Journal of Solid–State Circuits, 23(5):1171–1175.
Field Maintenance Print Set, KA780–01–01 Rev. A.
Fisher et al., "Very Long Instruction Word Architectures and the ELI–512," ACM pp. 140–150 (1983).
Furber, VSLI RISC Architecture and Organization, Chapter 3: Commercial VLSI RISC, pp. 124–129, Marcel Dekker, Inc., 1989.
GB Patent Application 8233733, INMOS, Ltd. Microcomputer, filed Nov. 26, 1962.
GE 600 Series, publication.
Gershon, Preface, IBM Systems Journal 26(4):324–325.
Green et al., "A Perspective on Advanced Peer–to–Peer Networking," IBM Systems Journal, 1987; 26(4):414–428.
Grimes et al., "64–bit Processor, The Intel i860 64–Bit Processor: A General–Purpose CPU with 3D Graphics Capabilities", published by Intel, p. 85 (Jul. 1989).
Hansen, "A RISC Microprocessor with Integral MMU and Cache Interface," ICCD '86, pp. 145–148, 1986.
Hennessy et al., "Hardware/software Tradeoff for Increased Performance," Technical Report No. 22.8, Computer Systems Laboratory, Feb. 1983, 24 pages.
Hennessy et al., "The MIPS Machine", COMPCON, IEEE, Spring 1982, pp. 2–7.
Hennessy, "Performance Issues in VLSI Processor Design," IEEE on VLSI in Computers, pp. 153–156 (1983).
Hinton, 80960—Next Generation, Compcon Spring 89, IEEE, 13–18 (1989).
Hollingsworth et al., "The Fairchild Clipper: Instruction Set Architecture and Processor Implementation," Report No. UCB/CSD 87/329, Computer Science Division (EECS), University of California Berkeley, California, (Feb. 11, 1987).
HP 9000 Instrument Controllers, Technical Specifications Guide, Oct. 1989.pdf.
HP 9000 Series Computer Systems, HP–UX Reference 09000–090004, Preliminary Nov. 1982.
Hughes, "Off–Chip Module Clock Controller", Delphion, IBM Technical Disclosure Bulletin, Sep. 1989.
Hunter, "Introduction to the Clipper Architecture," IEEE Micro, pp. 6–26 (Aug. 1987).
IBM RT PC, BYTE 1986 Extra Edition, Inside The IBM PCs, pp. 60–78.
INMOS Limited, IMS T424 Transputer Reference Manual, 1984.
Intel 388Tm DX Microprocessor 32–Bit CHMOS Microprocessor With Integrated Memory Management (1995).
Intel 80960CA User's Manual published by Intel (1989).
Intel Architecture Optimization Manual, Order No. 242816–003, published by Intel (1997).
Intel Architecture Software Developer's Manual, vol. 1; Basic Architecture, published by Intel (1997).
Intel 8080A/8080A–1/8080A–2, 8–Bit N–Channel Microprocessor, Order No. 231453–001, Its Respective Manufacturer (Nov. 1986).
Johnson et al., "A Variable Delay Line PLL for CPU–Coprocessor Synchronization," IEEE Journal of Solid–State Circuits, 23(5): 1218–1223, Oct. 1988.
Katevenis et al., "Reduced Instruction Set Computer Architecture for VLSI," Doctoral Dissertation, Oct. 1983.
Katevenis et al., "The RISC II Micro–Architecture," Journal of VLSI and Computer Systems, 1(2):138–152 (1984).
Kipp, "Micron Technology Inc. Reports Financial Results," Business Wire, New York, Sep. 26, 1988.
Kohn et al., "Introducing Intel i860 64–Bit Microprocessor," Intel Corporation, IEEE Micro (Aug. 1989).
Koopman, "RTX 4000," Proceedings of 1989 Rochester Forth Conference, pp. 84–86.
Koopman, "The WISC Concept: A proposal for a writable instruction set computer," BYTE, pp. 187–193. (Apr. 1987).
Koopman, Jr. et al. "MVP Microcoded CPU/16 Architecture," Proceedings of 1986 Rochester Forth Conference, pp. 277–280.
Koopman, Jr. et al., "WISC Technologies, Inc., Writable Instruction Set, Stack Oriented Computers: The WISC Concept," 1987 Rochester Forth Conference, Journal of Forth Application and Research, 5(1):49–71.
Koopman, Jr., Slack Computers: the new wave, 1989.
Loucks et al., "Advanced interactive Executive (AIX) Operating System Overview," IBM Systems Journal, 1987; 26(4):326–345.
Matick, "Self–Clocked Cache," Delphion, IBM Technical Disclosure Bulletin, Apr. 1985.
Miller, Frequency Modulated Ring Oscillator for a Mode Regulated Substrate Bias Generator, Delphion, IBM Technical Disclosure Bulletin, Sep. 1989.
Mills et al, "Box Structured Information Systems," IBM Systems Journal, 1987; 26(4):395–413.

MMP Portfolio, News Release: Roland Becomes 50th Licensee, Setting a Major Milestone in Moore Microprocessor Patent Licensing Program, 3 pages (May 1, 2009).

Moussouris et al., "A CMOS RISC Processor Integrated System Functions," Proceedings of 31st IEEE Computer Society International Conference, Cathedral Hill Hotel, San Francisco, CA Mar. 3–6, 1986, pp .126–131, 1986.

Olson, Semiconductor Die with Wiring Skirt (Packaging Structure), Delphion, IBM Technical Disclosure Bulletin, Jul. 1978.

O'Neil, "Pipeline Memory System for Drams", Delphion, IBM Technical Disclosure Bulletin, May 1989.

Patterson et al., "RISC I: A Reduced Instruction Set VLSI Computer," Proceedings of the 8th annual symposium on Computer Architecture on Computer Architecture, Miinneapolis, Minnesota, pp. 443–457 (May 1981).

Pountain, "The Archimedea A310," BYTE, 1987.

Przybyiski et al., "Organization and VLSI Implementation of MIPS," Technical Report CSL–TR–84–259, Apr. 1984.

Przybyiski, "The Design Verification and Testing of MIPS", 1984 Conference on Advanced Research in VLSI, pp. 100–109.

Roche et al., "Method of Assuring a Two–Cycle Start, Zero Cycle Stop, Non–Chopping on Chip Clock Control Throughout a VLSI Clock System," Delphion, IBM Technical Disclosure Bulletin, Sep. 1989.

Rowen et al., "A Pipelined 32b NMOS Microprocessors and Microcontrollers," IEEE International Solids–State Circuits Conference, pp. 180–181, 1984.

Rubinfeld et al., "The CVAX CPU, A CMOS VAX Microprocessor Chip", International Conference on Computer Design, Oct. 1987.

Sanamrad et al., "A Hardware System Analysis Processor," IEEE, Aug. 1987, pp. 73–80.

Shih, "Microprogramming Heritage of RISC Design," Proceedings of the 23rd annual workshop and symposium on Microprogramming and microarchitecture, pp. 275–280 (1990).

Sultan et al., "Implementing System–36 Advanced Peer–to–Peer Networking," IBM Systems Journal, 1987; 26(4):429–452.

Thornton, J. E., "Considerations in Computer Design Leading Up To the Control Data 6600," Control Data Chippewa Laboratory (1970).

VAX 11/780 Architecture Handbook vol. 1, 1977–1978, 2–7 and G–8.

VAX 8800 System Technical Description vol. 2, EK–KA881–TD–PRE, Section 6, Instruction Box (IBOX), Preliminary Edition (Jul. 1986).

VAX Maintenance Handbook; VAX–11/780, EK–VAXV2–HB–002, 1983 Edition.

VL86C010 RISC Family Data Manual, Application Specific Logic Product Division, 1987.

Waters et al., "AIX Usability Enhancements and Human Factors," IBM Systems Journal, 1987; 26(4):383–394.

Williams, "Chip Set Tackles Laptop Design Issues, Offers Flat–Panel VGA Control," Computer Design, Oct. 15, 1988; 27(19):21–22.

IEEE Std 796–1983, Microcomputer System Bus, pp. 9–46 (Dec. 1983).

Mead & Conway, Introduction to VLSI Systems, pp. 1–429 (1980).

EX PARTE REEXAMINATION CERTIFICATE ISSUED UNDER 35 U.S.C. 307

THE PATENT IS HEREBY AMENDED AS INDICATED BELOW.

Matter enclosed in heavy brackets [ ] appeared in the patent, but has been deleted and is no longer a part of the patent; matter printed in italics indicates additions made to the patent.

ONLY THOSE PARAGRAPHS OF THE SPECIFICATION AFFECTED BY AMENDMENT ARE PRINTED HEREIN.

Column 17, lines 12–37:

Most microprocessors derive all system timing from a single clock. The disadvantage is that different parts of the system can slow all operations. The microprocessor **50** provides a dual-clock scheme as shown in FIG. **17**, with the CPU **70** operating **[**a synchronously**]** *asynchronously* to I/O interface **432** forming part of memory controller **118** (FIG. **2**) and the I/O interface **432** operating synchronously with the external world of memory and I/O devices. The CPU **70** executes at the fastest speed possible using the adaptive ring counter clock **430**. Speed may vary by a factor of four depending upon temperature, voltage, and process. The external world must be synchronized to the microprocessor **50** for operations such as video display updating and disc drive reading and writing. This synchronization is performed by the I/O interface **432**, speed of which is controlled by a conventional crystal clock **434**. The interface **432** processes requests for memory accesses from the microprocessor **50** and acknowledges the presence of I/O data. The microprocessor **50** fetches up to four instructions in a single memory cycle and can perform much useful work before requiring another memory access. By decoupling the variable speed of the CPU **70** from the fixed speed of the I/O interface **432**, optimum performance can be achieved by each. Recoupling between the CPU **70** and the interface **432** is accomplished with handshake signals on lines **436**, with data/addresses passing on bus **90**, **136**.

AS A RESULT OF REEXAMINATION, IT HAS BEEN DETERMINED THAT:

Claims **3**–**5** and **8** are cancelled.

Claims **1**, **6** and **10** are determined to be patentable as amended.

Claims **2**, **7** and **9**, dependent on an amended claim, are determined to be patentable.

New claims **11**–**16** are added and determined to be patentable.

1. A microprocessor system, comprising a single integrated circuit including a central processing unit and an entire ring oscillator variable speed system clock in said single integrated circuit and connected to said central processing unit for clocking said central processing unit, said central processing unit and said ring oscillator variable speed system clock each including a plurality of electronic devices correspondingly constructed of the same process technology with corresponding manufacturing variations, a processing frequency capability of said central processing unit and a speed of said ring oscillator variable speed system clock varying together due to said manufacturing variations and due to at least operating voltage and temperature of said single integrated circuit; an on-chip input/output interface connected to exchange coupling control signals, addresses and data with said central processing unit; and a second clock independent of said ring oscillator variable speed system clock connected to said input/output interface, *wherein a clock signal of said second clock originates from a source other than said ring oscillator variable speed system clock.*

6. A microprocessor system comprising:

a central processing unit disposed upon an integrated circuit substrate, said central processing unit operating at a processing frequency and being constructed of a first plurality of electronic devices;

an entire oscillator disposed upon said integrated circuit substrate and connected to said central processing unit, said oscillator clocking said central processing unit at a clock rate and being constructed of a second plurality of electronic devices, thus varying the processing frequency of said first plurality of electronic devices and the clock rate of said second plurality of electronic devices in the same way as a function of parameter variation in one or more fabrication or operational parameters associated with said integrated circuit substrate, thereby enabling said processing frequency to track said clock rate in response to said parameter variation; an on-chip input/output interface, connected between said central processing unit and an *off-chip* external memory bus, for facilitating exchanging coupling control signals, addresses and data with said central processing unit; and

an *off-chip* external clock, independent of said oscillator, connected to said input/output interface wherein said *off-chip* external clock is operative at a frequency independent of a clock frequency of said oscillator *and wherein a clock signal from said off-chip external clock originates from a source other than said oscillator.*

10. In a microprocessor system including a central processing unit, a method for clocking said central processing unit comprising the steps of:

providing said central processing unit upon an integrated circuit substrate, said central processing unit being constructed of a first plurality of transistors and being operative at a processing frequency;

providing an entire variable speed clock disposed upon said integrated circuit substrate, said variable speed clock being constructed of a second plurality of transistors;

clocking said central processing unit at a clock rate using said variable speed clock with said central processing unit being clocked by said variable speed clock at a variable frequency dependent upon variation in one or more fabrication or operational parameters associated with said integrated circuit substrate, said processing frequency and said clock rate varying in the same way relative to said variation in said one or more fabrication or operational parameters associated with said integrated circuit substrate;

connecting an **[**on chip**]** *on-chip* input/output interface between said central processing unit and an *off-chip* external memory bus, and exchanging coupling control signals, addresses and data between said input/output interface and said central processing unit; and

clocking said input/output interface using an *off-chip* external clock wherein said *off-chip* external clock is operative at a frequency independent of a clock frequency of said variable speed clock *and wherein a clock signal from said off-chip external clock originates from a source other than said variable speed clock.*

11. A microprocessor system, comprising a single integrated circuit including a central processing unit and an entire ring oscillator variable speed system clock in said single integrated circuit and connected to said central processing unit for clocking said central processing unit, said central processing unit and said ring oscillator variable speed system clock each including a plurality of electronic devices correspondingly constructed of the same process technology with corresponding manufacturing variations, a processing frequency capability of said central processing unit and a speed of said ring oscillator variable speed system clock varying together due to said manufacturing variations and due to at least operating voltage and temperature of said single integrated circuit; an on-chip input/output interface connected to exchange coupling control signals, addresses and data with said central processing unit; and a second clock independent of said ring oscillator variable speed system clock connected to said input/output interface, wherein said central processing unit operates asynchronously to said input/output interface.

12. The microprocessor system of claim 11, in which said second clock is a fixed frequency clock.

13. A microprocessor system comprising: a central processing unit disposed upon an integrated circuit substrate, said central processing unit operating at a processing frequency and being constructed of a first plurality of electronic devices;

an entire oscillator disposed upon said integrated circuit substrate and connected to said central processing unit, said oscillator clocking said central processing unit at a clock rate and being constructed of a second plurality of electronic devices, thus varying the processing frequency of said first plurality of electronic devices and the clock rate of said second plurality of electronic devices in the same way as a function of parameter variation in one or more fabrication or operational parameters associated with said integrated circuit substrate, thereby enabling said processing frequency to track said clock rate in response to said parameter variation;

an on-chip input/output interface, connected between said central processing unit and an off-chip external memory bus, for facilitating exchanging coupling control signals, addresses and data with said central processing unit; and

an off-chip external clock, independent of said oscillator, connected to said input/output interface wherein said off-chip external clock is operative at a frequency independent of a clock frequency of said oscillator and further wherein said central processing unit operates asynchronously to said input/output interface.

14. The microprocessor system of claim 13 wherein said one or more operational parameters include operating temperature of said substrate or operating voltage of said substrate.

15. The microprocessor system of claim 13 wherein said oscillator comprises a ring oscillator.

16. In a microprocessor system including a central processing unit, a method for clocking said central processing unit comprising the steps of:

providing said central processing unit upon an integrated circuit substrate, said central processing unit being constructed of a first plurality of transistors and being operative at a processing frequency;

providing an entire variable speed clock disposed upon said integrated circuit substrate, said variable speed clock being constructed of a second plurality of transistors;

clocking said central processing unit at a clock rate using said variable speed clock with said central processing unit being clocked by said variable speed clock at a variable frequency dependent upon variation in one or more fabrication or operational parameters associated with said integrated circuit substrate, said processing frequency and said clock rate varying in the same way relative to said variation in said one or more fabrication or operational parameters associated with said integrated circuit substrate;

connecting an on-chip input/output interface between said central processing unit and an off-chip external memory bus, and exchanging coupling control signals, addresses and data between said input/output interface and said central processing unit; and

clocking said input/output interface using an off-chip external clock wherein said off-chip external clock is operative at a frequency independent of a clock frequency of said variable speed clock, wherein said central processing unit operates asychronously to said input/output interface.

* * * * *

US005809336C2

(12) **EX PARTE REEXAMINATION CERTIFICATE** (7887th)

United States Patent
Moore et al.

(10) **Number: US 5,809,336 C2**
(45) **Certificate Issued: Nov. 23, 2010**

(54) **HIGH PERFORMANCE MICROPROCESSOR HAVING VARIABLE SPEED SYSTEM CLOCK**

(75) Inventors: **Charles H. Moore**, 410 Star Hill Rd., Woodside, CA (US) 94062; **Russell H. Fish, III**, Mt. View, CA (US)

(73) Assignee: **Charles H. Moore**, Incline Village, NV (US)

Reexamination Request:
No. 90/009,457, Aug. 24, 2009

Reexamination Certificate for:

Patent No.:	**5,809,336**
Issued:	**Sep. 15, 1998**
Appl. No.:	**08/484,918**
Filed:	**Jun. 7, 1995**

Reexamination Certificate C1 5,809,336 issued Dec. 15, 2009

Certificate of Correction issued May 22, 2007.

Related U.S. Application Data

(62) Division of application No. 07/389,334, filed on Aug. 3, 1989, now Pat. No. 5,440,749.

(51) **Int. Cl.**

G06F 7/76	(2006.01)
G06F 7/48	(2006.01)
G06F 12/08	(2006.01)
G06F 7/78	(2006.01)
G06F 9/30	(2006.01)
G06F 9/32	(2006.01)
G06F 15/76	(2006.01)
G06F 15/78	(2006.01)
G06F 7/52	(2006.01)
G06F 9/38	(2006.01)
G06F 7/58	(2006.01)

(52) **U.S. Cl.** **710/25**; 711/E12.02; 712/E9.016; 712/E9.028; 712/E9.046; 712/E9.055; 712/E9.057; 712/E9.058; 712/E9.062; 712/E9.078; 712/E9.08; 712/E9.081

(58) **Field of Classification Search** ..... None
See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

4,348,743	A	9/1982	Dozier
4,691,124	A	9/1987	Ledzius et al.
4,766,567	A	8/1988	Kato
4,853,841	A	8/1989	Richter
4,931,748	A	6/1990	McDermott et al.
5,809,336	A	9/1998	Moore et al.

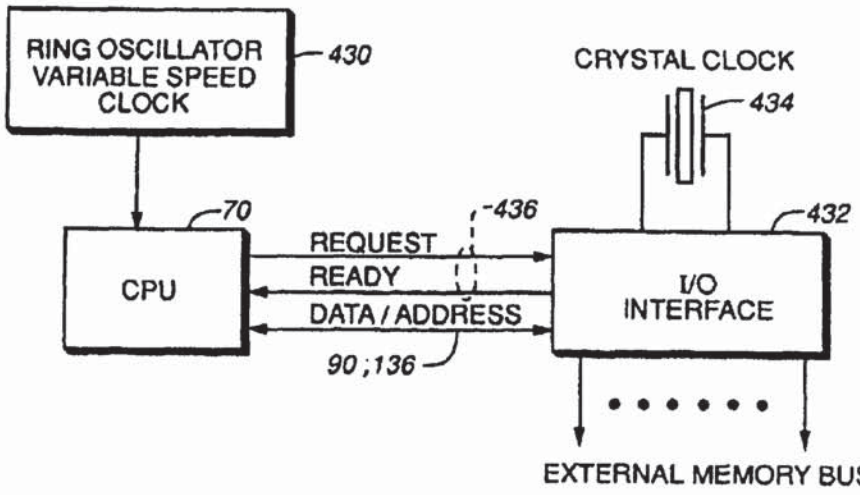
OTHER PUBLICATIONS

U.S. Patent No. 5,809,336, as certified by Ex Parte Reexamination Certificate (7235th) U.S. 5,809,336 C1, issued Dec. 15, 2009, 51 pages.
In re Recreative Technologies Corp., 83 F.3d 1394, 38 USPQ 2.d 1776 (Fed. Cir. 1996), 6 pages.

Primary Examiner—B. James Peikari

(57) **ABSTRACT**

A high performance, low cost microprocessor system having a variable speed system clock is disclosed herein. The microprocessor system includes an integrated circuit having a central processing unit and a ring oscillator variable speed system clock for clocking the microprocessor. The central processing unit and the ring oscillator variable speed system clock each include a plurality of electronic devices of like type, which allows the central processing unit to operate at a variable processing frequency dependent upon a variable speed of the ring oscillator variable speed system clock. The microprocessor system may also include an input/output interface connected to exchange coupling control signals, address and data with the central processing unit. The input/output interface is independently clocked by a second clock connected thereto.



EX PARTE
REEXAMINATION CERTIFICATE
ISSUED UNDER 35 U.S.C. 307

NO AMENDMENTS HAVE BEEN MADE TO THE PATENT

AS A RESULT OF REEXAMINATION, IT HAS BEEN DETERMINED THAT:

The patentability of claims **1**, **2**, **6**, **7** and **9-16** is confirmed.

Claims **3-5** and **8** were previously cancelled.

* * * * *

PROOF OF SERVICE

I hereby certify that, on this April 23, 2018, I electronically filed the foregoing Non-Confidential Brief of Appellants, with the Clerk of the Court using the CM/ECF system. Opposing Counsel listed below is ECF-registered and therefore has been served with the Non-Confidential Brief of Appellants.

Additionally, a printed copy of the Confidential version of the Brief of Appellants, has been served on the same date to the following:

Michael Eli Flynn-O'Brien, Esq. Steptoe & Johnson, LLP Steuart Tower 1 Market Street, Suite 1800 San Francisco, CA 94105 *Attorney for Appellees,* *Futurewei Technologies, Inc., Huawei Device Co., Ltd., Huawei Device USA Inc., Huawei Technologies Co., Ltd. and Huawei Technologies USA Inc.*	Jay Reiziss, Esq. McDermott Will & Emery LLP The McDermott Building 500 North Capitol Street NW Washington, DC 20001 *Attorney for Appellees,* *ZTE Corporation and* *ZTE USA, Inc.*
Mark D. Fowler, Esq. Erik Ryan Fuehrer, Esq. Aaron Wainscoat, Esq. DLA Piper US LLP 2000 University Avenue East Palo Alto, CA 94303-2248 *Attorneys for Appellees,* *Samsung Electronic Co., Ltd. and Samsung Electronics America, Inc.*	Christian A. Chu, Esq. Scott Andrew Elengold, Esq. Fish & Richardson, PC The McPherson Building 901 15th Street NW Washington, DC 20005 *Attorneys for Appellees,* *LG Electronics, Inc. and* *LG Electronics U.S.A., Inc.*
Stephen R. Smith, Cooley LLP 1299 Pennsylvania Avenue, NW, Suite 700 Washington, DC 20004 *Attorney for Appellees,* *Nintendo Co., Ltd. and* *Nintendo of America, Inc.*	

Upon acceptance by the Court of the e-filed document, six paper copies will be filed with the Court, via Federal Express, within the time provided in the Court's rules.

Dated: April 23, 2018

/s/ Denise M. De Mory
Denise De Mory (SBN 168076)

Counsel for Appellants/Plaintiffs,
TECHNOLOGY PROPERTIES LIMITED LLC,
PHOENIX DIGITAL SOLUTIONS LLC,
PATRIOT SCIENTIFIC CORPORATION

CERTIFICATE OF COMPLIANCE MOTIONS OR BRIEFS CONTAINING MATERIAL SUBJECT TO A PROTECTIVE ORDER

Briefs Containing Material Subject to a Protective Order

[X] This brief complies with the limitations set forth in Fed. Cir. R. 28(d) and contains [**1**] image (including numbers) marked as confidential, or

[ ] This brief does not comply with the word count limitations set forth in Fed. Cir. R. 27(m) and a motion is requesting permission to exceed the maximum word count

Dated: April 23, 2018

/s/ Denise M. De Mory
Denise De Mory (SBN 168076)

Counsel for Appellants/Plaintiffs,
TECHNOLOGY PROPERTIES LIMITED LLC,
PHOENIX DIGITAL SOLUTIONS LLC,
PATRIOT SCIENTIFIC CORPORATION

CERTIFICATE OF COMPLIANCE

This brief complies with the type-volume limitation of Federal Rule of Federal Circuit Rule 32(a). This brief contains 13,953 words, excluding the parts of the brief exempted by Federal Rule of Appellate Procedure 32(f) and Federal Circuit Rule 32(b).

This brief complies with the typeface requirements of Federal Rule of Appellate Procedure 32(a)(5) and the type style requirements of Federal Rule of Appellate Procedure 32(a)(6). The brief has been prepared in a proportionally spaced typeface using Microsoft Word 2016 in 14-point Times New Roman.

Dated: April 23, 2018

/s/ Denise M. De Mory
Denise De Mory (SBN 168076)

Counsel for Appellants/Plaintiffs,
TECHNOLOGY PROPERTIES LIMITED LLC,
PHOENIX DIGITAL SOLUTIONS LLC,
PATRIOT SCIENTIFIC CORPORATION