

**IN THE UNITED STATES DISTRICT COURT
FOR THE WESTERN DISTRICT OF TEXAS
AUSTIN DIVISION**

MOSAID TECHNOLOGIES INC.,

Plaintiff,

v.

INFINEON TECHNOLOGIES AG and
INFINEON TECHNOLOGIES AMERICAS
CORP.,

Defendants.

Civil No. 1:25-cv-00358

JURY TRIAL DEMANDED

COMPLAINT FOR PATENT INFRINGEMENT

Plaintiff MOSAID Technologies Inc. (“Plaintiff” or “MOSAID”) files this Complaint for Patent Infringement against Defendants Infineon Technologies AG (“Infineon AG”) and Infineon Technologies Americas Corp. (“Infineon Americas”) (collectively, “Defendants” or “Infineon”) alleging as follows:

NATURE OF SUIT

1. This is a claim for patent infringement arising under the patent laws of the United States, Title 35 of the United States Code.

PARTIES, JURISDICTION, AND VENUE

2. This action arises under the patent laws of the United States, 35 U.S.C. § 101, *et seq.* This Court has jurisdiction over this action pursuant to 28 U.S.C. §§ 1331 and 1338(a).

I. MOSAID

3. MOSAID (f/k/a Conversant Intellectual Property Management, Inc.) is a Canadian company having a principal place of business at 515 Legget Drive, Suite 100, Ottawa, ON, Canada.

4. MOSAID was founded in 1975 by engineers Richard Foss and Robert Harland. MOSAID focused its early efforts on inventing ways to improve Dynamic Random Access

Memory (DRAM) semiconductor chips. MOSAID's engineers also designed and built test equipment for debugging prototype memory chips.

5. MOSAID is now a leading patent management company with a world-class licensing team and an enviable record of success. MOSAID's success is defined by the numerous licenses it has signed with industry leaders. MOSAID prides itself on its patent expertise, a determination to engage in meaningful negotiation, and a commitment to transparency and the principled enforcement of high-quality patents.

6. MOSAID owns approximately 800 active patents as of February 2025. A large portion of the portfolio was developed in-house from research and development efforts focusing on flash memory technologies. MOSAID's microcomponents patents also include marquee patents relating to power management, based on various inventions derived from MOSAID's own R&D and product development businesses.

7. MOSAID is the assignee and owns all right, title, and interest to United States Patent No. 7,111,179 ("the '179 Patent"), United States Patent No. 7,051,306 ("the '306 Patent"), and United States Patent No. 7,349,448 ("the '448 Patent"). The '179 Patent, '306 Patent, and '448 Patent are collectively referred to herein as the "Asserted Patents."

II. INFINEON

8. Defendant Infineon Technologies AG ("Infineon AG") is a corporation organized and existing under the laws of the Federal Republic of Germany, and is located at Am Campeon 1-15, 85579 Neubiberg, Germany.

9. Infineon AG is listed on the Frankfurt Stock Exchange (ticker symbol: IFX) and in the USA on the over-the-counter market OTCQX International Premier (ticker symbol: IFNNY).

10. On information and belief, with a global presence, Infineon AG operates through its subsidiaries in the U.S.

11. Defendant Infineon Technologies Americas Corp. (“Infineon Americas”) is a corporation duly organized and existing under the laws of the state of Delaware and may be served with process through its Texas registered agent, Corporation Service Company d/b/a CSC – Lawyers Incorporating Service Company, 211 E. 7th Street, Suite 620, Austin, Texas 78701. Infineon Americas has a regular and established place of business in the Western District of Texas, including at 5204 E. Ben White Blvd, Austin, Texas 78741. On information and belief, Infineon Americas also maintains other offices in the State of Texas, including in Houston, Texas.

12. On information and belief, Infineon Americas is a wholly owned subsidiary of Infineon AG.

13. Cypress Semiconductor Corporation (“Cypress”) was an American semiconductor design and manufacturing company offering products such as NOR flash memories, F-RAM and SRAM Traveo microcontrollers, PSoCs, and others. On or around April 16, 2020, Infineon AG acquired 100% ownership of Cypress. On information and belief, at the time of the acquisition, Cypress was registered to do business in Texas, was transacting business in Texas and in this District, and maintained a regular and established place of business in this District at 5204 E. Ben White Blvd, Austin, Texas 78741. On information and belief, at least Infineon Americas took over Cypress’s operations at 5204 E. Ben White Blvd, Austin, Texas 78741 following Infineon AG’s acquisition of Cypress.

14. On information and belief, each Defendant, Infineon AG and Infineon Americas (collectively, “Infineon”), is a semiconductor company that designs, develops, makes, uses, offers for sale, sells in the United States, and/or imports into the United States, a variety of semiconductor products for the communications, Internet of Things (“IoT”), automotive, computer, and/or consumer electronics industries.

15. On information and belief, Infineon designs, develops, makes, uses, offers for sale, sells in the United States, and/or imports into the United States, certain processor products such as programmable system on a chip (“PSoC”) products including at least the PSoC 3, PSoC 4, PSoC 5, and PSoC 6 families, and wireless connectivity SoCs including at least the AIROC Wi-Fi and Bluetooth Combo products, among other substantially similar processor products (hereinafter, the “Accused Products”).

16. On information and belief, Infineon offers for sale and sells its products, including the Accused Products, directly to consumers in the United States, including in this District. For example, Infineon AG offers for sale and sells its products, including the Accused Products, in the United States via its website <https://www.infineon.com>:

We offer the following online ordering options

CY8C6347BZI-BLD33

Products > Microcontroller > 32-bit PSOC™ Arm® Cortex® Microcontroller > PSOC™ 6 Arm® Cortex®-M4 / M0+ > PSOC™ 63 Bluetooth™ LE MCU > CY8C6347BZI-BLD33

	Stock*	Price per unit	Quantity
 Infineon	140	\$ 9.01	<input type="text" value="1"/>
Standard Global Shipping		\$ 4.99	



(Exemplary PSoC 6 product available for purchase on Infineon’s website.)

We offer the following online ordering options

CY8C5266FNI-LP205T

Products > Microcontroller > 32-bit PSOC™ Arm® Cortex® Microcontroller > PSOC™ 5 LP Arm® Cortex®-M3 > CY8C52LPxxx > CY8C5266FNI-LP205T

	Stock*	Price per unit	Quantity
 Infineon	898	\$ 12.93	<input type="text" value="1"/>
Standard Global Shipping		\$ 4.99	

 Add to cart

(Exemplary PSoC 5 product available for purchase on Infineon's website.)

We offer the following online ordering options

CY8C4013LQI-411

Products > Microcontroller > 32-bit PSOC™ Arm® Cortex® Microcontroller > PSOC™ 4 Arm® Cortex®-M0/M0+ > CY8C4013LQI-411

	Stock*	Price per unit	Quantity
 Infineon	2442	\$ 1.63	<input type="text" value="1"/>
Standard Global Shipping		\$ 4.99	

 Add to cart

(Exemplary PSoC 4 product available for purchase on Infineon's website.)

We offer the following online ordering options

CY8C3244AXI-153

Products > Microcontroller > Legacy Microcontroller > Legacy 8-bit / 16-bit Microcontroller > PSoC™ 3 > CY8C32xxx > CY8C3244AXI-153

	Stock*	Price per unit	Quantity
 Infineon	99	\$ 7.39	<input type="text" value="1"/>
Standard Global Shipping		\$ 4.99	



(Exemplary PSoC 3 product available for purchase on Infineon's website.)

We offer the following online ordering options

CYW43439KUBGT

Products > Wireless Connectivity > AIROC™ Wi-Fi and Wi-Fi+Bluetooth® Combo > Wi-Fi 4 (802.11n) > CYW43439

	Stock*	Price per unit	Quantity
 Infineon	2459	\$ 5.72	<input type="text" value="1"/>
Standard Global Shipping		\$ 4.99	



(Exemplary AIROC Wi-Fi and Bluetooth Combo product available for purchase on Infineon's website.)

17. Infineon AG operates <https://www.infineon.com>, as stated in the website's footer and linked Terms of Use:



(<https://www.infineon.com>.)

Terms of Use

1. Scope

Any use of this web site provided by Infineon Technologies AG (hereinafter "Infineon") under the URL > <http://www.infineon.de> and/or > <http://www.infineon.com> (hereinafter "Web Site") is subject to these Terms of Use.

(<https://www.infineon.com/cms/en/about-infineon/usage-terms>.)

18. Infineon AG is also the Registrant Contact in the domain registration record for <https://www.infineon.com>.

19. Infineon also sells its Accused Products in the United States through Infineon Distribution Partners, including eCommerce Partners, Franchised Distributors, and Specialty Distributors.

20. On information and belief, Infineon sells its Accused Products to numerous companies in the United States, including companies in this District, for use in consumer end products.

21. On information and belief, Infineon AG has substantial connections to the State of Texas. For example, Infineon AG has sold products to companies in Texas, including in this District, for use in consumer end products. For example, Infineon's NOR Flash memory products

are used in Tesla vehicles, including at least the display system for the Tesla Model Y, which are manufactured in Austin, Texas at Tesla, Inc.'s global headquarters.

22. As another example, Infineon AG has partnered with Texas companies to provide products to customers in the United States, including in this District. For example, in 2023, Infineon AG announced its partnership with Spark Connected to provide a 500W wireless charging solution named Yeti. On information and belief, Spark Connected is a company organized under the laws of Texas and headquartered in Dallas, Texas. The Yeti 500W is a ready-to-integrate wireless charging module intended for the powering and charging of industrial machinery, autonomous mobile robots, automated guided vehicles, light electric vehicles, e-Mobility and other power-intensive applications. The module integrates at least Infineon's dual-core PSoC 63 Bluetooth Low Energy MCU, a microcontroller unit within Infineon's PSoC 6 product family, which infringes at least the '306 Patent.

23. In addition, on information and belief, Infineon AG has been collaborating with Siemens Industry Software Inc. (d/b/a Siemens Digital Industries Software) since at least November 19, 2024, to integrate Siemens' automotive embedded software platform with Infineon's AURIX microcontroller. Siemens Digital Industries Software is headquartered in Plano, Texas.

24. On information and belief, Infineon AG has showcased its products at trade shows in Texas, including at the Applied Power Electronics Conference ("APEC"). For example, Infineon AG showed several new products, including new packaging for its MOSFET family of products, at APEC 2022 held in Houston, Texas, and played a large role as a contributor to the conference program, participating in over 20 sessions. Infineon AG also had an exhibit at the APEC 2018 conference in San Antonio, Texas, where Infineon AG showcased its power

semiconductor technologies and participated in 15 sessions and seminars. In addition, Infineon AG attended the APEC 2011 conference in Fort Worth, Texas, where Infineon AG highlighted its OptiMOS 60-150V in CanPAK product. Infineon AG also showed its families of OptiMOS 3 power semiconductors and other products at the APEC 2008 conference held in Austin, Texas. At the APEC 2006 conference held in Dallas, Texas, Infineon AG introduced its OptiMOS 2 100V family of MOSFET devices and its second-generation silicon carbide Schottky diodes. In addition, at APEC 2005 held in Austin, Texas, Infineon AG introduced its CoolMOS CS Server series of high-performance power transistors.

25. Infineon AG is subject to specific and general personal jurisdiction in this Court. This Court has personal jurisdiction over Infineon AG because, on information and belief, Infineon AG has engaged in continuous, systematic, and substantial activities within this State, including substantial marketing and sales of products and services within this State and District. Furthermore, on information and belief, this Court has personal jurisdiction over Infineon AG because Infineon AG has committed acts of infringement giving rise to MOSAID's claims for patent infringement within and directed to this District.

26. On information and belief, Infineon AG has conducted, and does conduct, substantial business in this District, directly and/or through subsidiaries, agents, representatives, or intermediaries, including, but not limited to: (i) at least a portion of the acts of infringement alleged herein; (ii) purposefully and voluntarily placing one or more Accused Products into the stream of commerce with the expectation that they will be purchased by consumers in this District; and/or (iii) regularly doing or soliciting business, engaging in other persistent courses of conduct, or deriving substantial revenue from goods and services provided to individuals in this State and

in this District. Thus, Infineon AG is subject to this Court's specific and general personal jurisdiction pursuant to due process and the Texas Long-Arm Statute.

27. To the extent Infineon AG is not subject to jurisdiction in any state's courts of general jurisdiction, this Court has personal jurisdiction over Infineon AG pursuant to Federal Rule of Civil Procedure 4(k)(2) because MOSAID's claims arise under federal law and exercising jurisdiction is consistent with the United States Constitution and laws.

28. The exercise of personal jurisdiction over Infineon AG would not offend traditional notions of fair play and substantial justice.

29. Infineon AG has acknowledged that this Court has personal jurisdiction over it. *See, e.g., Staktek Group, L.P. v. Infineon Technologies AG*, No. 1:03-cv-00219-LY (W.D. Tex.) (Dkt. No. 3). Infineon AG has also admitted that personal jurisdiction existed over it in cases filed in other Texas district courts. *See, e.g., Third Dimension Semiconductor, Inc. v. Infineon Technologies North America Corp. & Infineon Technologies AG*, No. 6:08-cv-00129-LED (E.D. Tex.) (Dkt. No. 24).

30. Infineon AG has also previously filed suit in this District. *See, e.g., Infineon Technologies AG v. Harthcock*, No. 1:10-cv-00316-LY (W.D. Tex.) (Dkt. No. 1); *Infineon Technologies AG v. Infineon, L.L.C.*, No. 1:07-cv-00039-SS (W.D. Tex.) (Dkt. No. 1).

31. Under 28 U.S.C. §§ 1391(b)-(d) and 1400(b), venue is proper in this District as to Infineon AG at least because Infineon AG is a foreign corporation subject to personal jurisdiction in this District and has committed acts of infringement within this District giving rise to this action.

32. To the extent that there is no district in which an action may otherwise be brought against Infineon AG, venue is proper in this District as to Infineon AG under 28 U.S.C. § 1391(b)(3) because Infineon AG is subject to personal jurisdiction in this Court.

33. On information and belief, Infineon Americas has substantial connections to the State of Texas, including this District. For example, Infineon Americas has a regular and established place of business in this District at 5204 E. Ben White Blvd, Austin, Texas 78741. This Austin location is a Production site for Infineon products.

34. Infineon operates its largest North American semiconductor factory, also known as “Fab25,” in Austin, Texas. Infineon’s Fab25 employs more than 1,000 people and plays a critical role in the global semiconductor supply chain, producing up to one billion semiconductor chips per year for major automotive, industrial, and communications companies throughout the world. On information and belief, the Infineon products manufactured at Fab25 include high-speed NOR Flash memory products and custom microcontrollers that include one or more of the Accused Products.

35. On information and belief, a number of Infineon’s Austin employees are former Cypress employees who joined Infineon through the Cypress acquisition in 2020. On information and belief, these Austin employees include engineers, managers, technicians, and/or other personnel who work on research, development, design, manufacturing, testing, support, marketing, and/or sales of Accused Products.

36. On information and belief, Infineon Americas operates Fab25.

37. On information and belief, Infineon AG plays a role in the operation of Fab25, including because Infineon AG represents itself to the public as having a semiconductor manufacturing facility in Austin, Texas. For example, in March 2021, Infineon AG issued a press release from Munich stating that “Infineon Technologies (FSE: IFX / OTCQX: IFNNY) continues ramping up its manufacturing facility in Austin, Texas.” In addition, in May 2022, Infineon AG issued a press release stating “Infineon Technologies AG (FSE: IFX / OTCQX: IFNNY), today

announced that it has switched the operation of its Austin, Texas semiconductor factory, also known as ‘Fab25,’ to 100 percent renewable power.”

38. Infineon Americas is subject to specific and general personal jurisdiction in this Court. This Court has personal jurisdiction over Infineon Americas because it has engaged in continuous, systematic, and substantial activities within this State, including substantial research, development, production, marketing, and/or sales of products and services within this State and District. Furthermore, on information and belief, this Court has personal jurisdiction over Infineon Americas because Infineon Americas has committed acts of infringement giving rise to MOSAID’s claims for patent infringement within and directed to this District.

39. On information and belief, Infineon Americas has conducted and does conduct substantial business in this District, directly and/or through subsidiaries, agents, representatives, or intermediaries, such substantial business including, but not limited to: (i) at least a portion of the acts of infringement alleged herein; (ii) purposefully and voluntarily placing one or more Accused Products into the stream of commerce with the expectation that they will be purchased by consumers in this forum; and/or (iii) regularly doing or soliciting business, engaging in other persistent courses of conduct, or deriving substantial revenue from goods and services provided to individuals in Texas and in this District. Thus, Infineon Americas is subject to this Court’s specific and general personal jurisdiction pursuant to due process and the Texas Long-Arm Statute.

40. The exercise of personal jurisdiction over Infineon Americas would not offend traditional notions of fair play and substantial justice.

41. On information and belief, Infineon Americas has committed acts of infringement in this District and has regular and established places of business within this District under

28 U.S.C. § 1400(b). Thus, venue is proper in this District as to Infineon Americas under 28 U.S.C. § 1400(b).

42. Infineon maintains a permanent physical presence within this District. For example, Infineon maintains regular and established places of business at 5204 E. Ben White Blvd, Austin, Texas 78741.

43. On information and belief, Infineon's location(s) in this District are regular and established places of business under 28 U.S.C. § 1391, 28 U.S.C. § 1400(b), and *In re Cray, Inc.*, 871 F.3d 1355, 1360 (Fed. Cir. 2017).

a. On information and belief, Infineon's location(s) in this District are physical, geographical locations in this District. Each office location comprises one or more buildings or office spaces from which the business of Infineon is carried out. On information and belief, the location(s) are set apart for the purpose of carrying out Infineon's business, including, but not limited to, making, using, selling, offering for sale, and/or supporting Accused Products. On information and belief, Infineon advertises its physical location(s) in this District as places of its business.

b. On information and belief, Infineon's location(s) in this District are regular and established. Infineon identifies on its website (<https://www.infineon.com/cms/en/about-infineon/company/find-a-location/>) its address in this District as a regular and established place of Infineon's business.

c. On information and belief, Infineon's location(s) in this District are places of business of Infineon. On information and belief, Infineon conducts business from its location(s) in this District, including but not limited to, making, using, selling, offering for sale, and/or supporting Accused Products.

d. On information and belief, Infineon's location(s) in this District are physical, geographical location(s) in this District from which Infineon carries out its business.

e. On information and belief, Infineon employees work at Infineon's location(s) in this District. On information and belief, these Infineon employees are regularly and physically present at Infineon's location(s) during business hours and conduct Infineon's business while working there.

FACTUAL ALLEGATIONS

I. ASSERTED PATENTS

A. THE '179 PATENT

44. United States Patent No. 7,111,179 ("the '179 Patent") is entitled "Method and Apparatus for Optimizing Performance and Battery Life of Electronic Devices Based on System and Application Parameters." The United States Patent and Trademark Office duly and legally issued the '179 Patent on September 19, 2006, from U.S. Patent Application No. 10/268,914, filed on October 11, 2002.

45. The '179 Patent claims the priority benefit of U.S. Provisional Application No. 60/333,535, filed October 11, 2001.

46. MOSAID is the current owner of all rights, title, and interest in and to the '179 Patent, including the right to sue for past damages.

47. A true and correct copy of the '179 Patent is attached hereto as **Exhibit A** and is incorporated by reference herein.

48. The '179 Patent generally relates to an electronic device that adjusts power consumption in accordance with an analysis of parameters and events occurring over one or more time-periods. The electronic device monitors microprocessor, operating system, peripheral, and/or

device-level events and adjusts run-time parameters, such as microprocessor clock frequency and voltage, to reduce power consumption with minimal perceived degradation in performance.

49. Claim 1 of the '179 Patent is directed to:

1. A system for dynamically managing power within an electronic device comprising:

a power management device including:

a sampling module to ascertain and store a plurality of parameter values associated with said electronic device during at least one sampling interval;

an analysis module to combine said plurality of parameter values to produce an overall parameter value and to evaluate said overall parameter value to determine at least one control value to adjust power consumption of said electronic device in response to said parameter evaluation, wherein said plurality of parameters are arranged into a series of successive stages with each stage including at least one parameter, and said analysis module includes:

a normalization module to normalize said parameter values relative to a numeric range; and

a combination module to mathematically combine said normalized parameters of each subordinate stage with a succeeding stage to produce an overall value to determine said at least one control value to adjust said power consumption of said electronic device; and

an adjustment module to adjust power consumption of said electronic device in accordance with said at least one control value to control electronic device power consumption.

50. The '179 Patent solves a technological problem relating to power management for processing systems or devices. For example, the '179 Patent explains that portable and autonomously powered electronic devices (*e.g.*, computer systems, telephones, PDAs, etc.) are becoming more and more sophisticated, while at the same time, becoming smaller. However, while

the technology for portable electronic devices has progressed rapidly, improvements in battery capacity and energy density have not been as rapid. As a result, a gap has developed between the processing capabilities of portable electronic devices and the battery life of those devices. This gap has constrained the “autonomy” of portable electronic devices, reducing the amount of time between battery recharge or replacement, and thus limiting the true portability of these devices. Often, designers of portable electronic devices must delicately balance battery parameters such as size, weight, and cost against the portability and operational characteristics of the device.

51. In an attempt to overcome these problems, computer electronics manufacturers have incorporated many features into their hardware and software designs to extend battery life. A feature that has been introduced is frequency and voltage adjustment for microprocessors. Since power consumption is related to the clock frequency of the microprocessor and to the corresponding applied voltage, reductions in both parameters can have an effect on overall power consumption of the device. However, reducing clock frequency may also reduce instruction-processing performance. The prior art attempted to overcome this problem through practices such as: shutting off or deactivating hardware components of the computer system when activity fell below a threshold for a certain period of time, reducing or increasing the clock frequency of a microprocessor depending on the level of computer system activity, or monitoring the types of activities that a computer system performs to determine whether clock frequency and voltage are to be reduced or increased. However, as the '179 Patent notes, such techniques suffered from several disadvantages, including limited power savings and/or power conversation states that occurred at inopportune times and consequently adversely affected performance.

52. The inventions of the '179 Patent overcome the aforementioned problems and provide several advantages. In particular, the inventions significantly reduce power consumption

(effectively increasing battery life by a factor of two or more), while not appreciably affecting electronic device performance. These power consumption reductions are achieved through a combination of optimizations, aggregated together and analyzed over plural time-periods. The optimizations are based on very low-level system parameters related to the microprocessor hardware and operating system software, and high-level parameters related to the application and end-user operating environment.

53. Infineon is not licensed to the '179 Patent and was not licensed during the six years preceding this action.

B. THE '306 PATENT

54. United States Patent No. 7,051,306 ("the '306 Patent") is entitled "Managing Power on Integrated Circuits Using Power Islands." The United States Patent and Trademark Office duly and legally issued the '306 Patent on May 23, 2006, from U.S. Patent Application No. 10/840,893, filed on May 7, 2004.

55. The '306 Patent claims the priority benefit of U.S. Provisional Application No. 60/468,742, filed May 7, 2003.

56. MOSAID is the current owner of all rights, title, and interest in and to the '306 Patent, including the right to sue for past damages.

57. A true and correct copy of the '306 Patent is attached hereto as **Exhibit B** and is incorporated by reference herein.

58. The '306 Patent generally relates to managing power in an integrated circuit using power islands. For example, the integrated circuit includes a plurality of power islands where power consumption is independently controlled within each of the power islands. A power manager determines a target power level for one of the power islands. The power manager then determines an action to change a consumption power level of the one of the power islands to the

target power level. The power manager performs the action to change the consumption power level of the one of the power islands to the target power level. Power control circuitry controls the power of the one of the power islands. The power islands may be delineated based on geographic factors or functional circuitry of the integrated circuit. The action performed may be selecting a clock frequency for one of the power islands or selecting a clock for one of the power islands, modifying voltage for one of the power islands, or powering on or off one of the power islands.

59. Claim 16 of the '306 Patent is directed to:

16. A system for an integrated circuit comprising a plurality of power islands where power consumption is independently controlled within each of the power islands, the system comprising:

power control circuitry configured to control power for one of the power islands; and

a power manager configured to determine a target power level for the one of the power islands based on needs and operation of the integrated circuit, determine at least one of actions to change a power consumption level of the one of the power islands to the target power level, and perform the at least one of the actions to change the power consumption level of the one of the power islands to the target power level where one of the actions comprises selecting a frequency for the one of the power islands.

60. The '306 Patent solves a technological problem relating to managing power on an integrated circuit. As the '306 Patent explains, one design goal for integrated circuits is to reduce power consumption. Devices with batteries such as cell phones and laptops especially need a reduction in power consumption in the integrated circuit to extend the charge of the battery, as well as prevent overheating and lower heat dissipation. However, as the '306 Patent notes, one problem with many integrated circuits in the prior art was that power consumption was not efficiently utilized. For example, the entire integrated circuit may operate at a maximum frequency just to support an application needing that maximum frequency, while other portions of the

integrated circuit can operate at a lower frequency. In another example, inactive circuitry within the integrated circuit consumes power and increases the likelihood of leakage. Inefficient power consumption may also adversely affect performance of the integrated circuit. With the growing complexity of integrated circuits, the reduction of power consumption is even more important when the integrated circuit employs more functionality. Systems-on-a-chip, for example, employ many different types of functionalities within a single chip and consume even more power than single function integrated circuits.

61. Some prior integrated circuits have employed voltage islands or multiple clocks to lower power consumption. One problem with these integrated circuits is that the voltages in the power island and the frequencies of the multiple clocks are static. The voltages and the frequencies do not dynamically change based on the needs and operation of the integrated circuit. The invention addresses the above problems by managing power in an integrated circuit using power islands, where power consumption is independently controlled within each of the power islands.

62. Infineon is not licensed to the '306 Patent and was not licensed during the six years preceding this action.

C. THE '448 PATENT

63. United States Patent No. 7,349,448 ("the '448 Patent") is entitled "Distributed Multiplexing Circuit with Built-In Repeater." The United States Patent and Trademark Office duly and legally issued the '448 Patent on March 25, 2008, from U.S. Patent Application No. 10/632,885, filed on August 1, 2003.

64. MOSAID is the current owner of all rights, title, and interest in and to the '448 Patent, including the right to sue for past damages.

65. A true and correct copy of the '448 Patent is attached hereto as **Exhibit C** and is incorporated by reference herein.

66. The '448 Patent generally relates to systems and methods for distributing multiplexor ("MUX") functionality. For example, generally, a distributed multiplexor circuit can select between multiple data connections, each of which is separated from multiplexor logic by some distance on an integrated circuit ("IC"). In such a system, each data connection transmits its data word to an associated distributed multiplexor logic stage. For example, a first stage logic is configured to receive a first data word from one of the connections and to transmit the first data word received. A second stage logic is configured to receive the first data word from the first stage logic and to select a selected data word between the first data word and a second data word received from another of the plurality of data connections based upon a set of select signals. The second stage logic is further configured to transmit the selected data word.

67. Claim 4 of the '448 Patent is directed to:

4. A multiplexor circuit, comprising:

a plurality of data connections;

first stage logic configured to receive a first data word from one of the connections and to transmit the first data word received based on a set of select signals; and

second stage logic configured to receive the first data word from the first stage logic and to select a selected data word between the first data word and a second data word received from another of the plurality of data connections based upon a set of select signals, the second stage logic configured to transmit the selected data word to a third stage logic and to increase a signal strength of the first data word when the first data word is transmitted, if the first data word is selected.

68. The '448 Patent solves a technological problem relating to routing congestion in MUX circuits such as those implemented on an IC and used to select a particular data connection from multiple connections. For example, a MUX circuit within an IC may comprise eight data connections that transmit 128-bit data words. Typically, each of the 128-bit data words may be

routed to multiplexor logic on the IC via a connection that comprises 128 routing wires, *e.g.*, a wire for each bit that is transmitted. Therefore, in this particular example, 1,024 routing wires on the IC would be used to route eight data words from eight data connections to the multiplexor logic. Such a MUX circuit was prone to routing congestion and the inherent problems associated therewith, *e.g.*, difficulty in designing and fabricating. Making matters worse, the number of routing wires increased as the number of data connections increased.

69. In the past, dynamic MUXs and tristate MUXs have been employed in an attempt to address routing congestion caused by large numbers of routing wires. However, a dynamic MUX typically requires more power, has smaller noise margins, and takes more time to design correctly. Further, usage of tristate MUXs could be acceptable when the distances between the data connections and the MUX logic are small, but speed and noise degradation typically increases as the distances between the data connections increase.

70. The '448 Patent discloses and claims staged multiplexer circuitry to improve MUX design and address the above problems. For example, as the '448 Patent explains, the distributed multiplexor circuitry includes distributed multiplexor logic ("DML") stages associated with each data connection as well as select logic. A DML stage receives a data word from an associated data connection as well as a data word from a previous DML stage. Select logic indicates which data word from the multiple data connections is outputted by transmitting a set of select signals to each DML stage. The select signals are indicative of which data word is to be selected for further transmission. In accordance with these select signals, each DML stage selects either the data word it receives from a previous DML stage, if any, or another data word it receives from another data connection. In the final stage of the plurality of consecutive stages, the distributed multiplexor

circuit drives the resulting selected data word indicated by the select logic onto a system bus or some other type of conductive connection.

71. Infineon is not licensed to the '448 Patent and was not licensed during the six years preceding this action.

II. DEFENDANTS' KNOWLEDGE OF THE ASSERTED PATENTS

72. Infineon had knowledge of the Asserted Patents prior to the filing of this suit.

73. Between at least November 2017 and August 26, 2024, MOSAID (f/k/a Conversant Intellectual Property Management) engaged in many discussions, including through emails, letters, and business and technical meetings, with Cypress and/or Infineon regarding their infringement of MOSAID's patents, including the Asserted Patents.

74. On or about November 3, 2017, MOSAID sent a letter and exemplary claim charts to Cypress exhibiting Cypress's infringement of several patents, including the '179, '306, and '448 Patents. MOSAID's November 3, 2017 exemplary claim charts included infringement assertions for at least Claim 1 of the '179 Patent, Claim 1 of the '306 Patent, and Claim 8 of the '448 Patent. Receipt was confirmed by Terence Woodsome, Cypress's then-Deputy General Counsel.

75. On January 24, 2018, MOSAID followed up with Cypress regarding Cypress's response to MOSAID's letter and with a request to schedule a face-to-face meeting to further discussions.

76. On April 17, 2018, MOSAID sent Cypress additional information and updated exemplary claim charts regarding Cypress's infringement of the '179, '306, and '448 Patents. MOSAID's updated exemplary claim chart for the '306 Patent included infringement assertions for at least Claim 16.

77. On June 29 and July 30, 2018, MOSAID followed up with Cypress regarding MOSAID's infringement allegations and again requested a face-to-face meeting.

78. On August 2 and August 22, 2018, MOSAID again contacted Cypress to request a face-to-face meeting to occur sometime in September 2018.

79. On September 26, 2018, MOSAID met with Cypress for further discussion. During the meeting, MOSAID shared additional information and updated exemplary claim charts regarding Cypress's infringement of at least the '179 and '306 Patents. MOSAID sent copies of the updated exemplary claim charts for the '179, '306, and '448 Patents to Cypress by email on October 2, 2018.

80. On February 27, 2019, MOSAID followed up with Cypress after having received no further responses from Cypress following the September 2018 meeting. In that February 27 correspondence, MOSAID also provided new exemplary claim charts for new infringement assertions and asked to continue discussions. The new exemplary claim charts included charts for the '306 Patent (which identified additional products accused of infringing the '306 Patent).

81. Between February 2019 and February 2020, MOSAID and Cypress exchanged numerous communications and materials regarding Cypress's infringement. MOSAID also requested another meeting to discuss any outstanding technical matters and present its thoughts on business terms for a license to MOSAID's patent portfolio.

82. The parties engaged in another year of back and forth between February 2020 and February 2021. By then, Infineon had completed its acquisition of Cypress, and Cypress's personnel, including Mr. Woodsome, became employees of Infineon and transitioned to using Infineon email addresses.

83. MOSAID and Infineon held meetings on February 24 and March 10, 2021. MOSAID indicated its desire to work on a formal license proposal based on the active patents still in discussion, which included the Asserted Patents.

84. On February 28, 2023, MOSAID reconnected with Infineon on the parties' outstanding discussions, and for the next year the parties exchanged further communications regarding Infineon's infringement of the Asserted Patents.

85. In February and April 2024, MOSAID noted that technical discussions between MOSAID and Infineon (including Cypress) had been going on for many years, so MOSAID was now at a point where it would appreciate knowing whether Infineon was ready and willing to take a license or provide feedback on MOSAID's business proposal.

86. On May 23, 2024, MOSAID again reached out to Infineon in an effort to license Infineon to the Asserted Patents, and requested "business discussions to finally resolve these issues." However, Infineon ignored MOSAID's request and did not respond.

87. Ultimately, since at least November 3, 2017, MOSAID and Infineon (including Cypress) have engaged in numerous technical and business discussions relating to a license for MOSAID's patent portfolio, including each of the Asserted Patents. During those discussions, MOSAID offered to provide a license to the Asserted Patents. But to date, Infineon has not made any offer to license MOSAID's patent portfolio, including the Asserted Patents. Nor has Infineon ceased its infringing conduct.

88. Before filing this lawsuit, on August 26, 2024, MOSAID attempted one last time to resolve this dispute and suggested "a call so that we don't misunderstand your position on this matter." MOSAID again offered to license its patent portfolio, including the Asserted Patents, to Infineon on reasonable terms. But again, Infineon ignore MOSAID's correspondence.

III. THE ACCUSED PRODUCTS

89. Infineon manufactures, uses, offers for sale, and/or sells in the United States, and/or imports into the United States, PSoCs and wireless connectivity SoCs, including the Accused Products, for use in a variety of end-products in the communications, IoT, automotive, computer,

and/or consumer electronics industries. Both the Accused Products and the consumer end-products incorporating the Accused Products have been, and continue to be, widely available for sale in the United States.

90. Infineon's Accused Products include at least the Accused Products, and substantially similar products, made, used, sold, or offered for sale in the United States, and/or imported into the United States.

91. Infineon has advertised, offered for sale, and sold—and continues to advertise, offer for sale, and sell—the Accused Products on Infineon's website.

92. Infineon advertises, offers for sale, and sells the accused PSoC 3 product family on its website, including for example at <https://www.infineon.com/cms/en/product/microcontroller/legacy-microcontroller/legacy-8-bit-16-bit-microcontroller/psoc-3/>, and as shown above in paragraph 16.

93. Infineon advertises, offers for sale, and sells the accused PSoC 4, PSoC 5, and PSoC 6 product families on its website, including for example at <https://www.infineon.com/cms/en/product/microcontroller/32-bit-psoc-arm-cortex-microcontroller/>, and as shown above in paragraph 16.

94. Infineon advertises, offers for sale, and sells accused wireless connectivity products on its website, including for example at <https://www.infineon.com/cms/en/product/wireless-connectivity/airoc-wi-fi-plus-bluetooth-combos/>, and as shown above in paragraph 16.

COUNT I: INFRINGEMENT OF THE '179 PATENT

95. MOSAID incorporates by reference and realleges paragraphs 1 through 94 as if specifically set forth herein.

96. In violation of 35 U.S.C. § 271(a), Infineon has directly infringed one or more of the claims of the '179 Patent, including at least Claim 1, either literally and/or under the doctrine

of equivalents, by making, using, selling, and/or offering for sale in the United States, and/or importing into the United States, without authority, at least one of the Accused Products.

97. Claim 1 of the '179 Patent recites:

1. A system for dynamically managing power within an electronic device comprising:

a power management device including:

a sampling module to ascertain and store a plurality of parameter values associated with said electronic device during at least one sampling interval;

an analysis module to combine said plurality of parameter values to produce an overall parameter value and to evaluate said overall parameter value to determine at least one control value to adjust power consumption of said electronic device in response to said parameter evaluation, wherein said plurality of parameters are arranged into a series of successive stages with each stage including at least one parameter, and said analysis module includes:

a normalization module to normalize said parameter values relative to a numeric range; and

a combination module to mathematically combine said normalized parameters of each subordinate stage with a succeeding stage to produce an overall value to determine said at least one control value to adjust said power consumption of said electronic device; and

an adjustment module to adjust power consumption of said electronic device in accordance with said at least one control value to control electronic device power consumption.

98. The Accused Products, including at least the PSoC 3, PSoC 4, and PSoC 5 families, practice each element of Claim 1 of the '179 Patent.

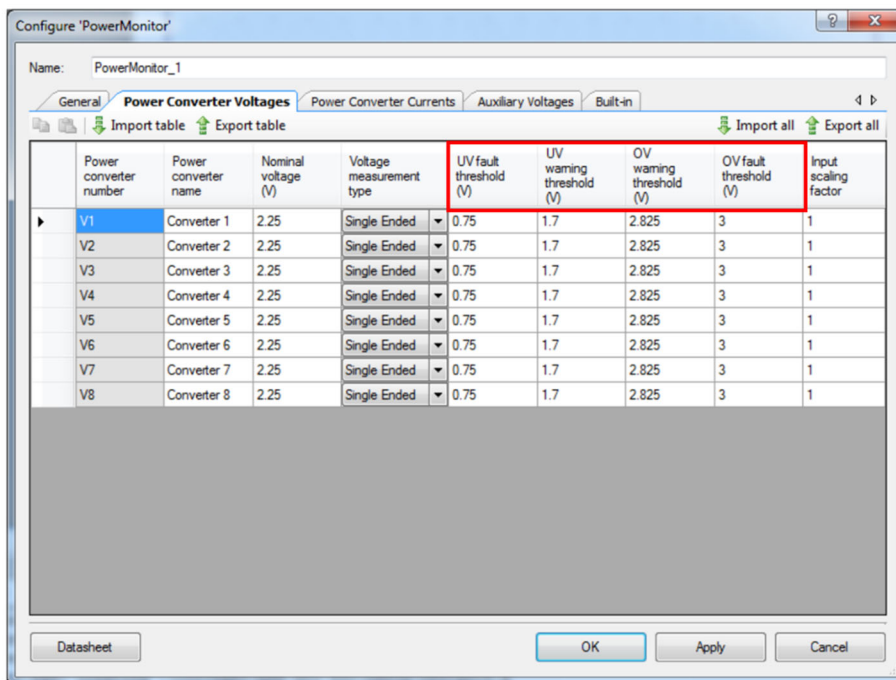
99. The Accused Products include a power management device. For example, at least the PSoC 3, 4, and 5 include a Power Monitor component that monitors the health of power converters and is capable of generating warnings and faults based on user-defined thresholds.

100. The power management device includes a sampling module to ascertain and store a plurality of parameter values associated with said electronic device during at least one sampling interval. For example, the Power Monitor is configured to measure power converter output voltages and supports averaging of both voltage and current measurements. The average value is calculated as a running average, which produces a new average with each scan that is the average of the previous N scans.

101. The power management device also includes an analysis module to combine said plurality of parameter values to produce an overall parameter value and to evaluate said overall parameter value to determine at least one control value to adjust power consumption of said electronic device in response to said parameter evaluation, wherein said plurality of parameters are arranged into a series of successive stages with each stage including at least one parameter. For example, the Power Monitor is capable of combining the sampled voltage parameter values into an overall running average. The average value is calculated as a running average, which produces a new average with each scan. The analysis module uses the average overall voltage value to evaluate fault and warning thresholds.

Power Converter Voltages Tab

This enables the user to describe the power converter voltages in the system. The figure below shows the voltage tab when Number of Converters is set to 8 in the General Tab.



*(Power Monitor 1.20 PSoC® Creator Component Datasheet, at 12.)
(annotation in red added)*

102. The analysis module includes a normalization module to normalize said parameter values relative to a numeric range. For example, at least the PSoC 3, 4, and 5 can be configured with a voltage scaling factor (*e.g.*, between 0.001 and 1) for a specified power converter that controls the amount of attenuation applied to the converter output voltage external to the PSoC.

103. The analysis module also includes a combination module to mathematically combine said normalized parameters of each subordinate stage with a succeeding stage to produce an overall value to determine said at least one control value to adjust said power consumption of said electronic device. For instance, on information and belief, at least the PSoC 3, 4, and 5 mathematically combine the normalized voltage values as a running average for use when evaluating faults or warning conditions.

104. In addition, the Accused Products include an adjustment module to adjust power consumption of said electronic device in accordance with said at least one control value to control electronic device power consumption. For example, at least the PSoC 3, 4, and 5 include a voltage sequencer connectable to the Power Monitor that supports sequencing and monitoring of up to 32 power converter rails.

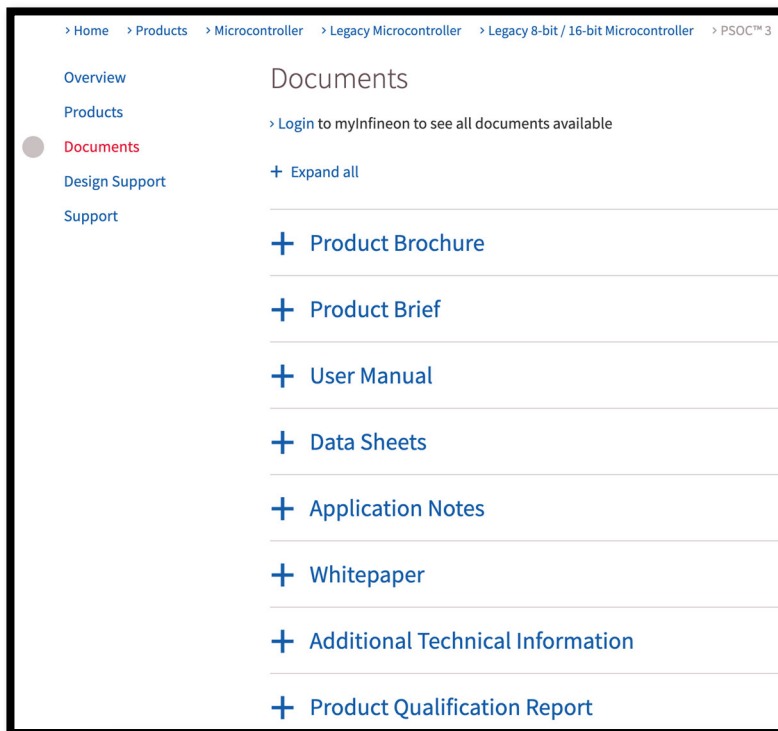
105. During license discussions, MOSAID provided Infineon with exemplary claim charts explaining in detail Infineon's infringement of the '179 Patent, including at least Claim 1.

106. In violation of 35 U.S.C. § 271(b), Infineon has infringed one or more of the claims of the '179 Patent, including at least Claim 1, indirectly by inducing infringement by third parties, including for example Infineon's customers and/or end-users of the Accused Products, including at least the PSoC 3, 4, and 5, in this District and elsewhere in the United States. For example, on information and belief, at least the PSoC 3, 4, and 5 have been used by Infineon's customers and/or end-users in consumer products including at least wearables, hearables, fitness products, mobile devices, automotives, IoT devices, home automation, home appliances, small kitchen appliances, coffee machines, washing machines, and industrial control systems. Direct infringement by Infineon's customers and/or end-users occurred at least by the use of the Accused Products, including at least the PSoC 3, 4, and 5, including use of consumer products incorporating them.

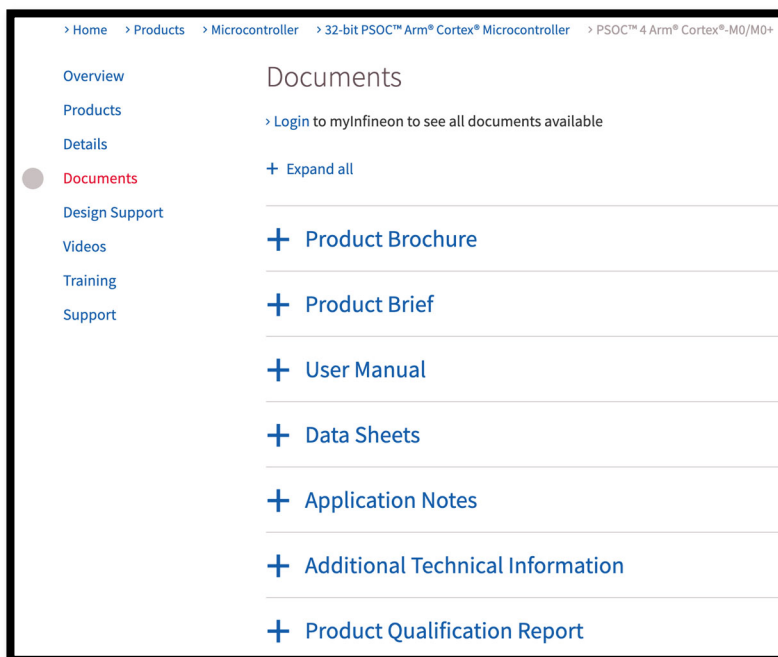
107. On information and belief, Infineon supplied hardware, firmware, and/or software that were especially made or especially adapted to practice the inventions claimed in the '179 Patent, including at least Claim 1, to induce third parties, including for example Infineon's customers and/or end-users of the Accused Products, including at least the PSoC 3, 4, and 5, to use such products in a manner that would infringe one or more claims of the '179 Patent, including at least Claim 1.

108. On information and belief, Infineon marketed and advertised the Accused Products, including at least the PSoC 3, 4, and 5, including on its website, to induce third parties, including Infineon's customers and/or end-users, to use the products in a manner that would infringe one or more claims of the '179 Patent, including at least Claim 1. *See, e.g.*, <https://www.infineon.com/cms/en/product/microcontroller/legacy-microcontroller/legacy-8-bit-16-bit-microcontroller/psoc-3>; <https://www.infineon.com/cms/en/product/microcontroller/32-bit-psoc-arm-cortex-microcontroller/psoc-4-32-bit-arm-cortex-m0-mcu>; <https://www.infineon.com/cms/en/product/microcontroller/32-bit-psoc-arm-cortex-microcontroller/32-bit-psoc-5-lp-arm-cortex-m3>; <https://www.infineon.com/cms/en/design-support/tools/sdk/psoc-software/psoc-3-5-components/power-monitor>; <https://www.infineon.com/cms/en/design-support/tools/sdk/psoc-software/psoc-4-components/psoc-4-power-monitor>.

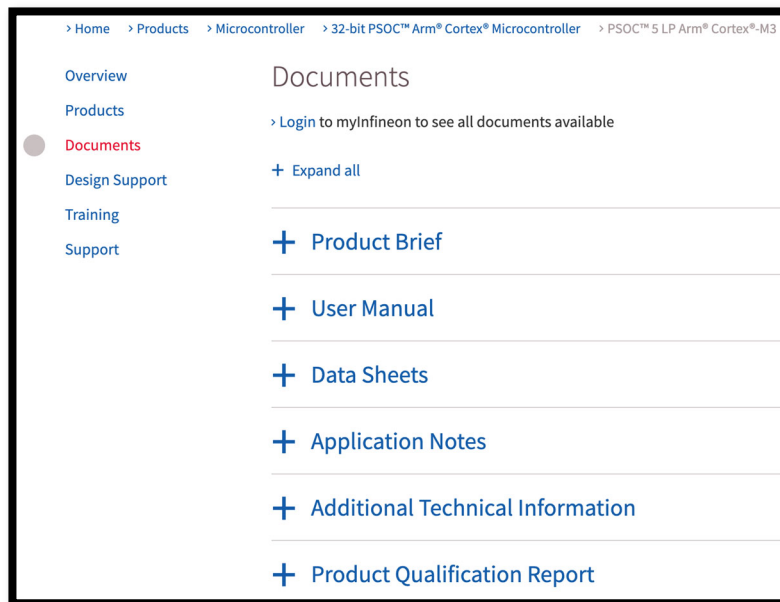
109. On information and belief, Infineon furnished instructive materials, technical support, and information concerning the operation and use of the Accused Products, including at least the PSoC 3, 4, and 5, to induce third parties, including Infineon's customers and/or end-users, to use the products in a manner that would infringe one or more claims of the '179 Patent, including at least Claim 1. For example, on its website, Infineon furnishes at least product brochures, product briefs, user manuals, data sheets, application notes, whitepapers, product qualification reports, and additional technical information such as architecture technical reference manuals:



(Exemplary categories of instructive materials provided for the PSoC 3, with additional materials provided for each product subcategory under their respective web subpages.)

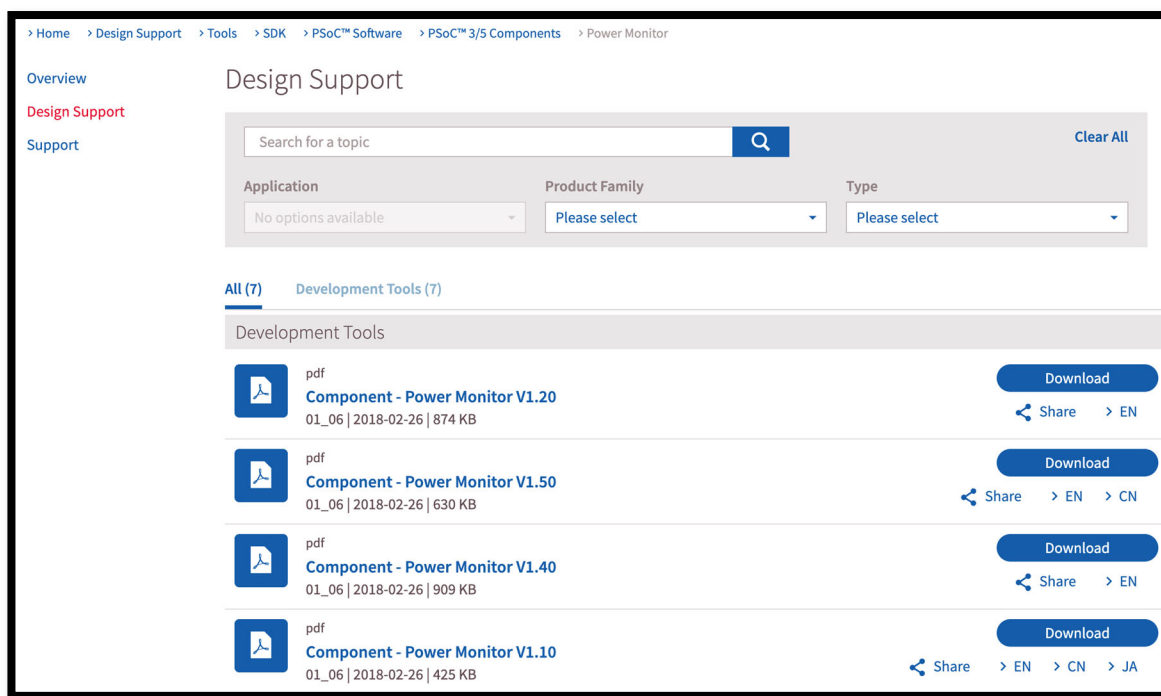


(Exemplary categories of instructive materials provided for the PSoC 4, with additional materials provided for each product subcategory under their respective web subpages.)

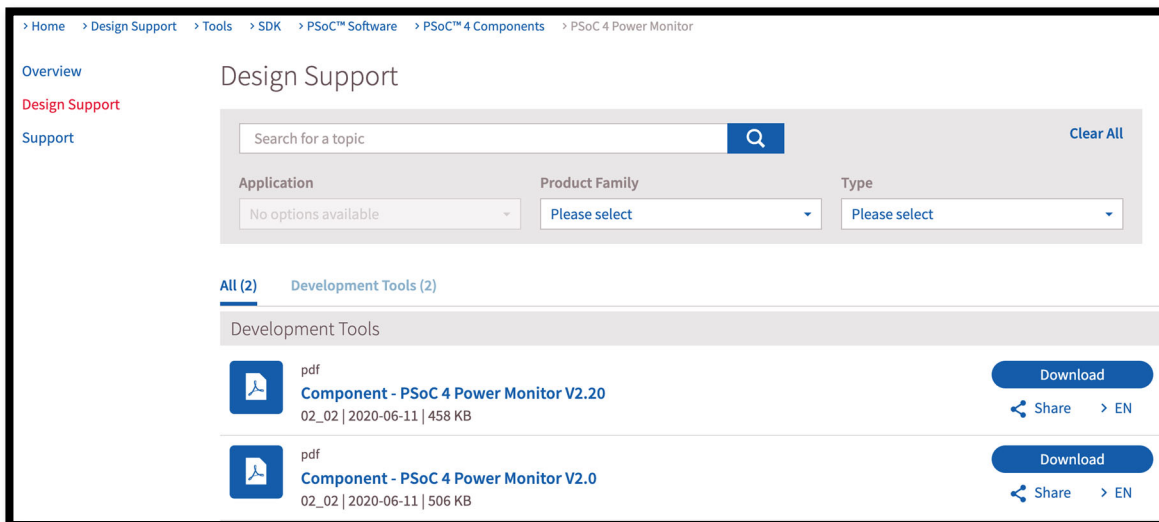


(Exemplary categories of instructive materials provided for the PSoC 5, with additional materials provided for each product subcategory under their respective web subpages.)

110. On its website, Infineon also furnishes development tools and design support materials for the Accused Products, including at least the PSoC 3, 4, and 5:



(Exemplary design support materials for the Power Monitor component for PSoC 3/5.)



(Exemplary design support materials for the Power Monitor component for PSoC 4.)

111. Further, Infineon provides APIs and sample firmware source code:

Application Programming Interface

Application Programming Interface (API) routines allow you to configure the component using software. The following table lists and describes the interface to each function. The subsequent sections cover each function in more detail.

By default, PSoC Creator assigns the instance name "PowerMonitor_1" to the first instance of a component in a given design. You can rename it to any unique value that follows the syntactic rules for identifiers. The instance name becomes the prefix of every global function name, variable, and constant symbol. For readability, the instance name used in the following table is "PowerMonitor."

Note: When using PSoC3 silicon, the user should create a Keil .cyre reentrancy file and should add the CyIntSetVector(), CyIntSetPriority(), PowerMonitor_PM_AMux_Current_Unset() and PowerMonitor_PM_AMux_Voltage_Unset() APIs in this file to avoid reentrancy related warnings during project compilation.

(Power Monitor 1.20 PSoC® Creator Component Datasheet, at 16.)

Sample Firmware Source Code

PSoC Creator provides numerous example projects that include schematics and example code in the Find Example Project dialog. For component-specific examples, open the dialog from the Component Catalog or an instance of the component in a schematic. For general examples, open the dialog from the Start Page or **File** menu. As needed, use the **Filter Options** in the dialog to narrow the list of projects available to select.

Refer to the "Find Example Project" topic in the PSoC Creator Help for more information.

(Power Monitor 1.20 PSoC® Creator Component Datasheet, at 36.)

Application Programming Interface

Application Programming Interface (API) routines allow you to configure the Component using software. The following sections list and describe each function and dependencies.

API functions allow configuration of the Component using the CPU.

By default, PSoC Creator assigns the instance name **PowerMonitor_P4_1** to the first instance of a Component in a given design. You can rename it to any unique value that follows the syntactic rules for identifiers. The instance name becomes the prefix of every global function name, variable, and constant symbol. For readability, the instance name used in the following table is **PowerMonitor_P4**.

Modules

- [Initialization](#)
General APIs are used for initialization of the Component with defaults taken from the Component customizer parameters.
- [General](#)
General APIs are used for run-time configuration of the Component during active power mode.
- [Interrupts](#)
Interrupt accessors.

(PSoC 4 Power Monitor 2.20 PSoC® Creator Component Datasheet, at 6.)

112. In addition, on its website, Infineon provides training materials and help from Infineon support engineers:

Training

Steering wheel hands-on detection with Infineon's CAPSENSE™ technology



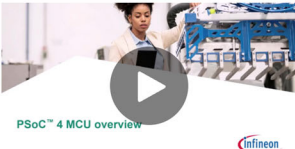
Steering wheel hands-on detection with Infineon's CAPSENSE™ technology

- Outline the basics of Infineon's CAPSENSE™ technology, as well as some application widgets and use cases
- Describe steering wheel hands-on detection implementation using PSoC™ 4 and CAPSENSE™ technology

[Watch eLearning](#)

PSoC™ 4 MCU Overview (E1)

[Share](#)

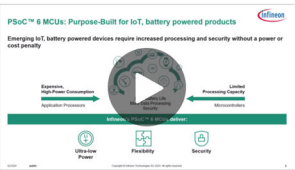


PSoC™ 4 MCU overview

PSoC™ 4 is a scalable and reconfigurable platform architecture for a family of mixed-signal programmable embedded system controllers with an Arm® Cortex®-M0+ CPU. It is a combination of a microcontroller with digital programmable logic, high-performance analog-to-digital conversion, opamps, comparators, a capacitive touch-sensing system (CAPSENSE™) and standard communication and timing peripherals.

PSoC™ 6 MCU Portfolio Overview (E1)

[Share](#)



PSoC™ 6 MCUs: Purpose-Built for IoT, battery-powered products

Emerging IoT, battery-powered devices require increased processing and security without a power or cost penalty

Expensive High-Power Consumption Applications Processors

Limited Processing Capacity Microcontrollers

Optimize Power, Flexibility, Security

This short introductory training provides a high level overview to our PSoC™ 6 microcontroller.

(Exemplary Training Materials.)



(Link to the Infineon Developer Community for the PSoC 4.)



(Link to the Infineon Developer Community for the PSoC 5.)

113. Infineon knew or should have known of the '179 Patent and its infringement through a series of correspondence and meetings expressly notifying Infineon of the '179 Patent and Infineon's infringement thereof. As a result, Infineon knew or should have known it infringed one or more claims of the '179 Patent, including at least Claim 1, at least as early as November 3, 2017, when MOSAID began discussions with Cypress regarding the '179 Patent and provided actual notice of infringement.

114. At a minimum, Infineon knew or should have known of the '179 Patent and its infringement no later than April 2020, when Infineon acquired Cypress and continued discussions with MOSAID regarding the '179 Patent and Infineon's infringement thereof.

115. Despite this knowledge, Infineon continued to induce third parties, including Infineon's customers and/or end-users of the Accused Products, including at least the PSoC 3, 4, and 5, to infringe one or more claims of the '179 Patent, including at least Claim 1, with the specific intent to cause infringement. Infineon knew or should have known that those acts would

induce actual infringement by third parties, including Infineon's customers and/or end-users of the Accused Products, including at least the PSoC 3, 4, and 5, of one or more of the claims of the '179 Patent, including at least Claim 1.

116. Therefore, Infineon has induced infringement by others of one or more of the claims of the '179 Patent, including at least Claim 1, with the specific intent to induce acts that constitute infringement of the '179 Patent and with knowledge that such acts infringed one or more claims of the '179 Patent, including at least Claim 1.

117. In violation of 35 U.S.C. § 271(c), Infineon has infringed one or more of the claims of the '179 Patent, including at least Claim 1, indirectly by contributing to infringement by third parties, including for example Infineon's customers and/or end-users of the Accused Products, including at least the PSoC 3, 4, and 5, in this District and elsewhere in the United States. Direct infringement by Infineon's customers and/or end-users occurred at least by the use of the Accused Products, including at least the PSoC 3, 4, and 5, including use of consumer products incorporating them.

118. On information and belief, Infineon made and sold hardware, firmware, and/or software components (*e.g.*, processors and/or software drivers) especially made or especially adapted to practice the invention claimed in the '179 Patent, including at least Claim 1. For example, as explained above, Infineon made and sold PSoC hardware, including the PSoC 3, 4, and 5 architectures, and accompanying APIs and firmware code. On information and belief, such hardware, firmware, and/or software components (i) are a material part of the invention and (ii) are not staple articles or commodities of commerce suitable for substantial non-infringing use at least because they are specifically designed to perform the claimed functionality. Any other use of such

hardware, firmware, and/or software would be unusual, far-fetched, illusory, impractical, occasional, aberrant, or experimental.

119. Therefore, Infineon has contributed to the infringement by others of one or more of the claims of the '179 Patent, including at least Claim 1.

120. Infineon's infringement of one or more claims of the '179 Patent, including at least Claim 1, was willful.

121. As explained above, Infineon had actual knowledge of the '179 Patent and its infringement thereof at least as early as November 3, 2017, when MOSAID began discussions with Cypress regarding the '179 Patent, and in any event no later than April 2020, when Infineon acquired Cypress and continued discussions with MOSAID.

122. As explained above, on several occasions beginning at least as early as November 3, 2017, MOSAID notified Infineon (including Cypress) of the '179 Patent and Infineon's infringement thereof, including infringement of at least Claim 1.

123. Despite knowing of the '179 Patent at least as early as November 3, 2017 (and no later than April 2020), Infineon did not cease its infringing activities. Infineon continued to infringe one or more claims of the '179 Patent, including at least Claim 1, in disregard of MOSAID's patent rights. As a result, Infineon deliberately and intentionally infringed the '179 Patent, and continued to do so, after receiving express and actual knowledge of both the '179 Patent and its infringement thereof.

124. Therefore, Infineon's infringement of the '179 Patent, including at least Claim 1, was willful, wanton, malicious, in bad faith, deliberate, consciously wrongful, flagrant, or characteristic of a pirate, entitling MOSAID to increased damages pursuant to 35 U.S.C. § 284 and to attorneys' fees and costs incurred in prosecuting this action pursuant to 35 U.S.C. § 285.

COUNT II: INFRINGEMENT OF THE '306 PATENT

125. MOSAID incorporates by reference and realleges paragraphs 1 through 124 as if specifically set forth herein.

126. In violation of 35 U.S.C. § 271(a), Infineon has directly infringed one or more of the claims of the '306 Patent, including at least Claim 16, either literally and/or under the doctrine of equivalents, by making, using, selling, and/or offering for sale in the United States, and/or importing into the United States, without authority, at least one of the Accused Products.

127. Claim 16 of the '306 Patent recites:

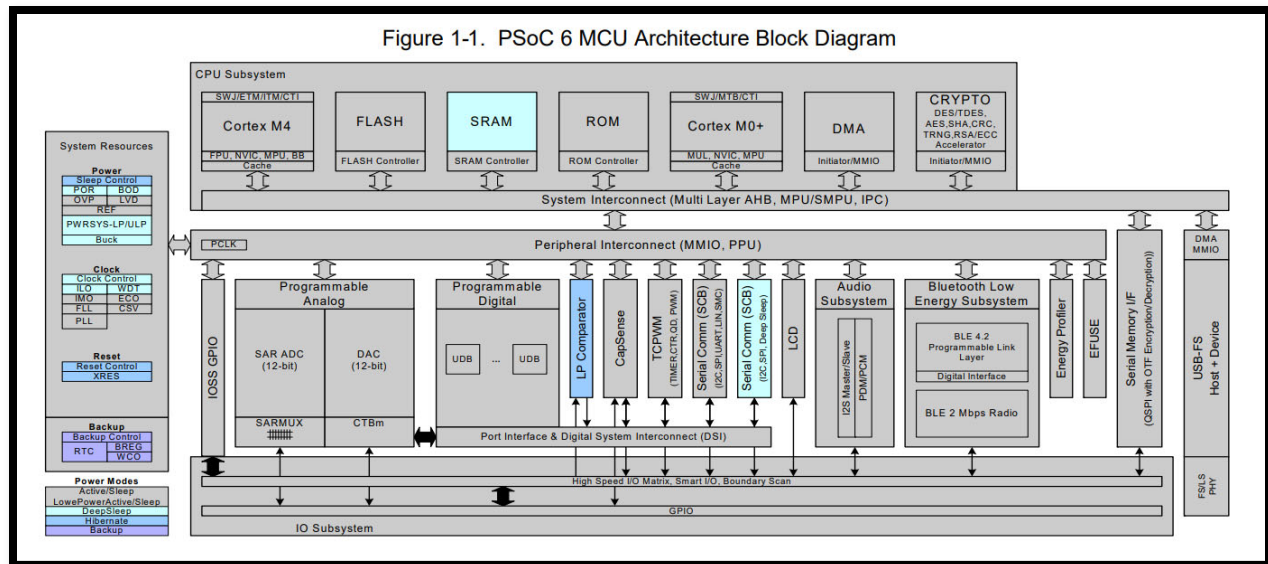
16. A system for an integrated circuit comprising a plurality of power islands where power consumption is independently controlled within each of the power islands, the system comprising:

power control circuitry configured to control power for one of the power islands; and

a power manager configured to determine a target power level for the one of the power islands based on needs and operation of the integrated circuit, determine at least one of actions to change a power consumption level of the one of the power islands to the target power level, and perform the at least one of the actions to change the power consumption level of the one of the power islands to the target power level where one of the actions comprises selecting a frequency for the one of the power islands.

128. The Accused Products, including at least the PSoC 6 and AIROC products, practice each element of Claim 16 of the '306 Patent.

129. The Accused Products include a system for an integrated circuit comprising a plurality of power islands where power consumption is independently controlled within each of the power islands. For example, the PSoC 63 Bluetooth®LE Line of processors (in Infineon's PSoC 6 family) include multiple power islands as illustrated in the following diagram.

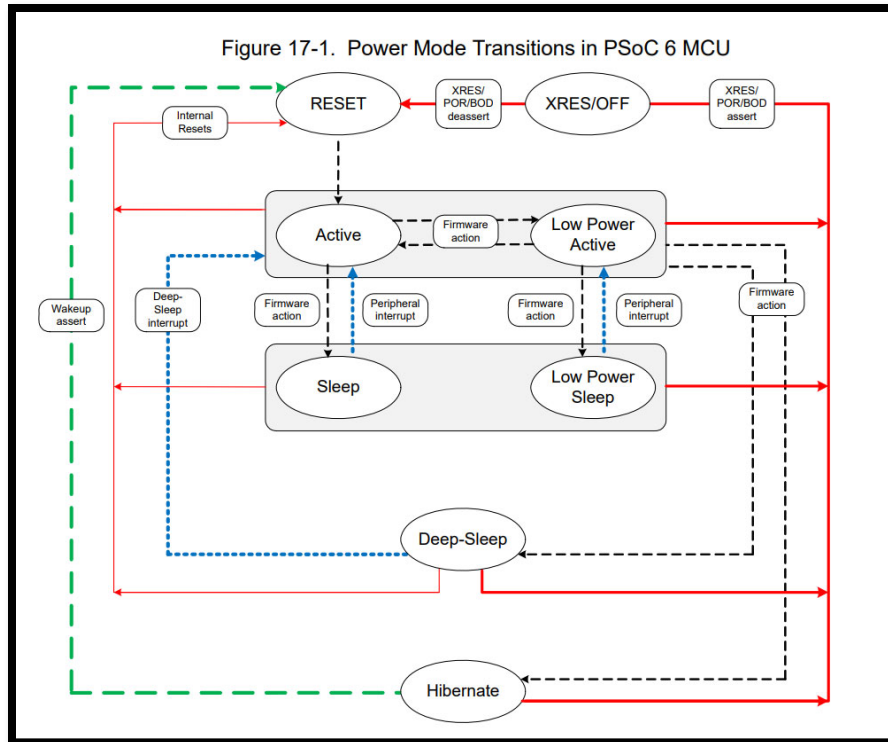


(PSoC 63 with BLE Architecture Technical Reference Manual (TRM), at 22.)

130. On the PSoC MCU, Cortex M-4 and Cortex-M0+ both support their own CPU sleep modes and each CPU can be in sleep independent of the state of the other CPU.

131. The Accused Products, including at least the PSoC 6 and AIROC products, include power control circuitry configured to control power for one of the power islands. For example, the PSoC 63 line includes circuitry for enabling and disabling clocks to peripherals, powering on and off clock sources, and powering on and off peripherals and parts inside the PSoC 6 MCU.

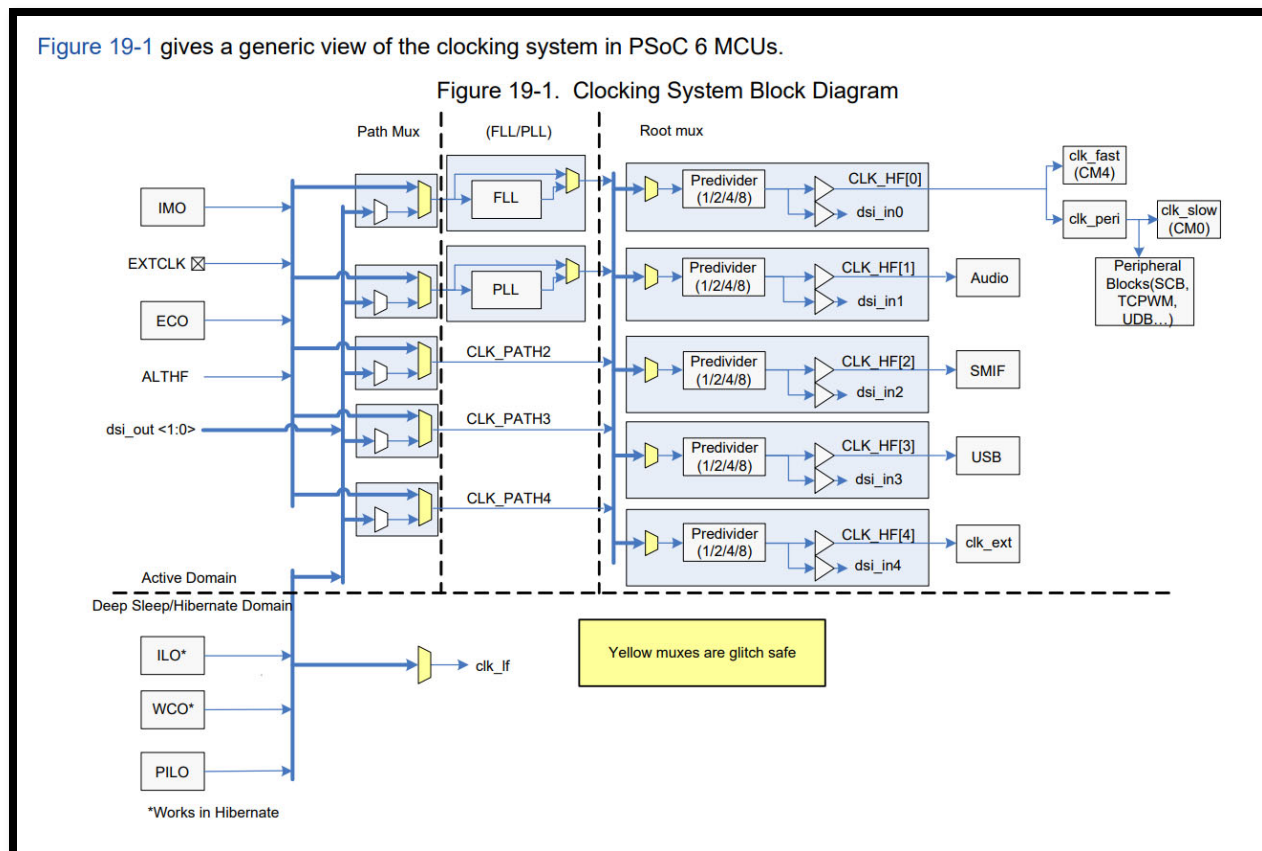
132. As illustrated in the below figure, the Accused Products, including at least the PSoC 6 and AIROC products, include a power manager configured to determine a target power level for the one of the power islands based on needs and operation of the integrated circuit. The firmware may decide to enable or disable specific peripherals and power domains depending on the application and power requirement.



(PSoC 63 with BLE Architecture Technical Reference Manual (TRM), at 137.)

133. The power manager is also configured to determine at least one of actions to change a power consumption level of the one of the power islands to the target power level and perform the at least one of the actions to change the power consumption level of the one of the power islands to the target power level where one of the actions comprises selecting a frequency for the one of the power islands. For example, the PSoC 6 MCU architecture supports dynamic voltage and frequency scaling. When the power manager transitions between power states, it will determine an action to change the power consumption level of a power island to the target power level, including by adjusting the frequency.

Figure 19-1 gives a generic view of the clocking system in PSoC 6 MCUs.



(PSoC 63 with BLE Architecture Technical Reference Manual (TRM), at 153.)

134. During license discussions, MOSAID provided Infineon with exemplary claim charts explaining in detail Infineon's infringement of the '306 Patent, including at least Claim 16.

135. In violation of 35 U.S.C. § 271(b), Infineon has infringed one or more of the claims of the '306 Patent, including at least Claim 16, indirectly by inducing infringement by third parties, including for example Infineon's customers and/or end-users of the Accused Products, including at least the PSoC 6 and AIROC products, in this District and elsewhere in the United States. For example, on information and belief, at least the PSoC 6 and AIROC products have been used by Infineon's customers and/or end-users in consumer products including at least IoT devices, such as wearables, smart rings, fitness trackers, eBikes, home appliances, home automation devices, smart home accessories, cordless power tools, industrial IoT devices, anomaly detection tags,

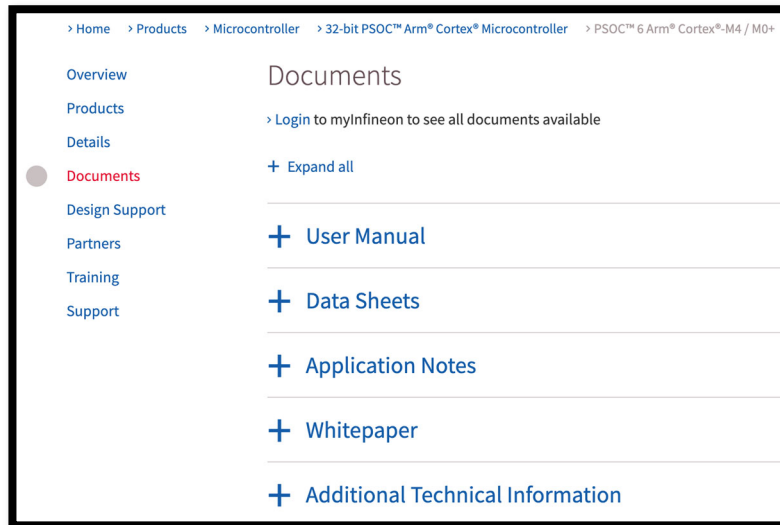
portable medical devices, and medical wearables such as vitals monitoring devices. Direct infringement by Infineon's customers and/or end-users occurred at least by the use of the Accused Products, including at least the PSoC 6 and AIROC products, including use of consumer products incorporating them.

136. On information and belief, Infineon supplied hardware, firmware, and/or software that were especially made or especially adapted to practice the inventions claimed in the '306 Patent, including at least Claim 16, to induce third parties, including for example Infineon's customers and/or end-users of the Accused Products, including at least the PSoC 6 and AIROC products, to use such products in a manner that would infringe one or more claims of the '306 Patent, including at least Claim 16.

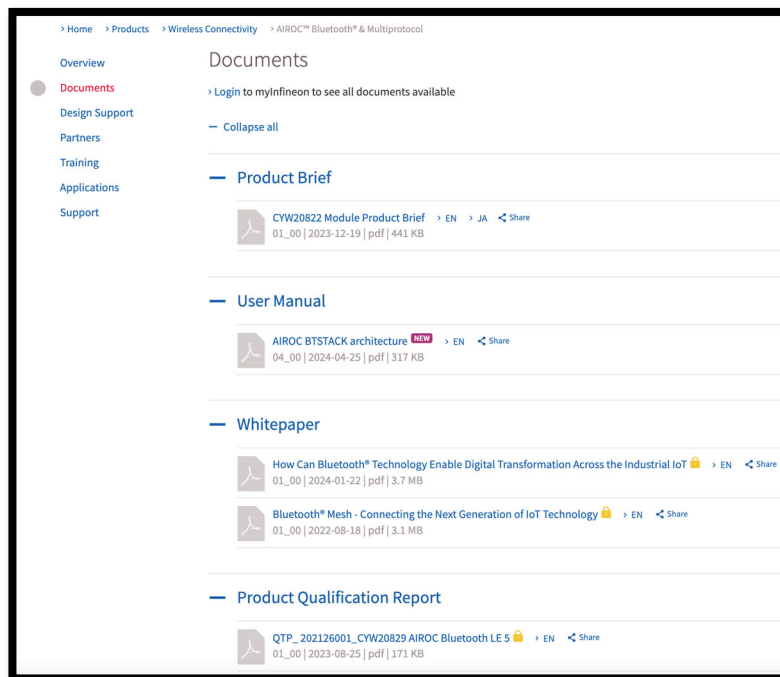
137. On information and belief, Infineon marketed and advertised the Accused Products, including at least the PSoC 6 and AIROC products, including on its website, to induce third parties, including Infineon's customers and/or end-users, to use the products in a manner that would infringe one or more claims of the '306 Patent, including at least Claim 16. *See, e.g.,* <https://www.infineon.com/cms/en/product/microcontroller/32-bit-psoc-arm-cortex-microcontroller/psoc-6-32-bit-arm-cortex-m4-mcu>; <https://www.infineon.com/cms/en/product/wireless-connectivity/airoc-bluetooth-le-bluetooth-multiprotocol>.

138. On information and belief, Infineon furnished instructive materials, technical support, and information concerning the operation and use of the Accused Products, including at least the PSoC 6 and AIROC products, to induce third parties, including Infineon's customers and/or end-users, to use the products in a manner that would infringe one or more claims of the '306 Patent, including at least Claim 16. For example, on its website, Infineon furnishes at least

product brochures, product briefs, user manuals, data sheets, application notes, whitepapers, product qualification reports, and additional technical information such as architecture technical reference manuals:

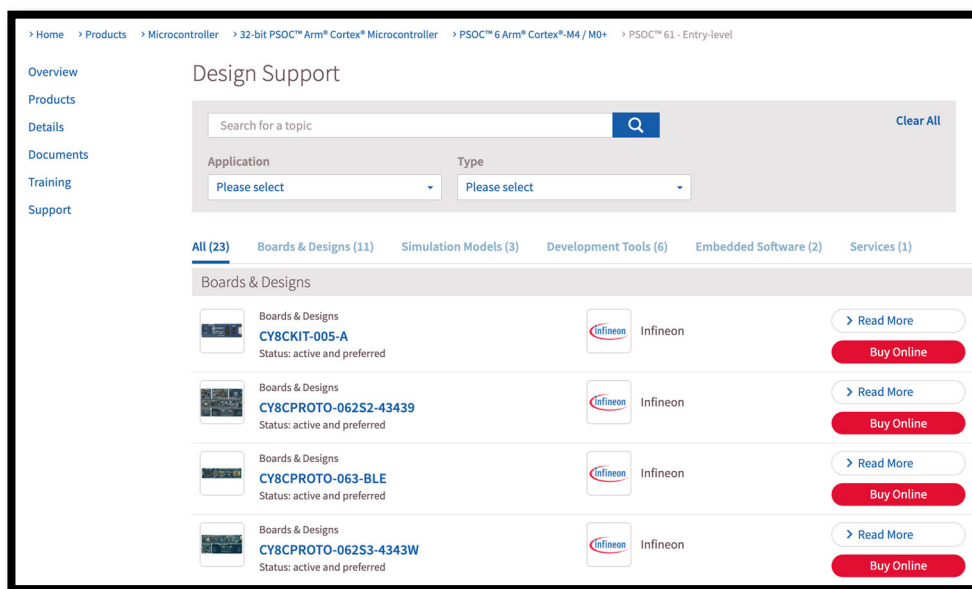


(Exemplary categories of instructive materials provided for the PSoC 6, with additional materials provided for each product subcategory under their respective web subpages.)

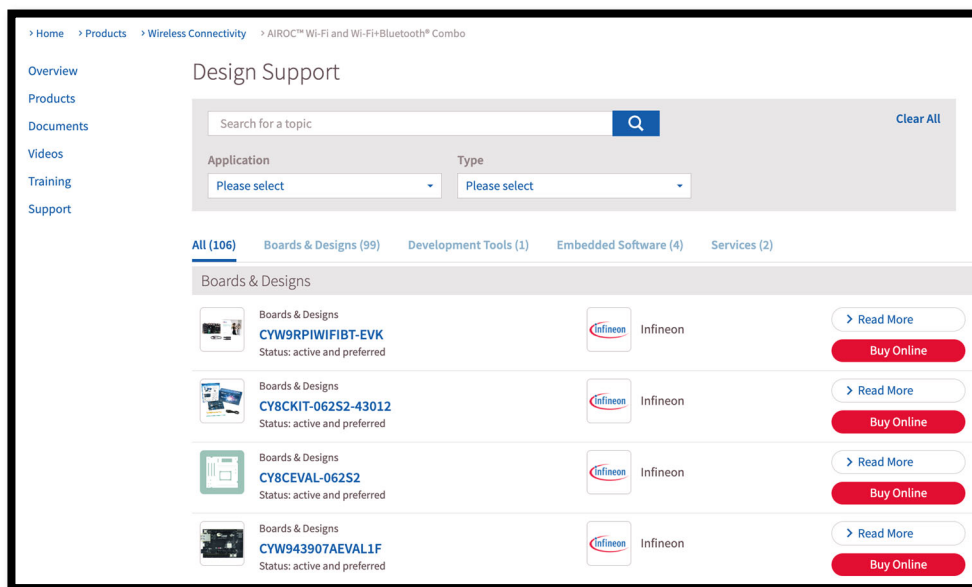


(Exemplary instructive materials provided for the AIROC products, with additional materials provided for each product subcategory under their respective web subpages.)

139. On its website, Infineon also furnishes development tools and design support materials for the Accused Products, including at least the PSoC 6 and AIROC products:



(Exemplary design support materials for PSoC 6.)



(Exemplary design support materials for AIROC products.)

140. Further, Infineon provides software, including code examples, drivers, and firmware:

[Home](#) > [Design Support](#) > [Software](#) > [Code Examples](#) > PSOC™ 6 Code Examples for PSOC™ Creator

PSOC™ 6 Code Examples for PSOC™ Creator

[Overview](#)
[Design Support](#)
[Support](#)

General Description

The code examples linked in the table below are compatible with PSOC™ Creator 4.2. In PSOC™ Creator use **File > Code Example** to download and import the example. These examples focus primarily on the PSOC™ 6 MCU. They are not compatible with ModusToolbox™ IDE.

PSOC™ 6 MCU Code Examples

CE#	Title	Description
CE212736	PSOC™ 6 MCU with Bluetooth Low Energy (BLE) Connectivity - Find Me	This code example demonstrates the implementation of a simple BLE immediate alert service (IAS)-based Find Me profile using PSOC™ 6 MCU with BLE Connectivity.
CE213903	PSOC™ 6 MCU Basic Bootloaders	These examples demonstrate basic bootloading with PSOC™ 6 MCU. This includes downloading an application from a host and installing it in device flash, and then transferring control to that application.
CE220120	PSOC™ 6 MCU: Blocking Mode Flash Write	This example demonstrates how to write to the flash memory of a PSOC™ 6 MCU device. In this example, the flash write API function blocks the caller until the write is completed.
CE219881	PSOC™ 6 MCU Switching Between Power Modes	This example demonstrates how to transition PSOC™ 6 between the following power modes - Active, Sleep, Low Power Active, Low Power Sleep and Deep Sleep.

(Exemplary code examples for PSoC 6.)

[Home](#) > [Design Support](#) > [Software](#) > [Device Driver & Libraries](#) > AIROC™ Wi-Fi & Bluetooth Linux and Android Drivers

[Overview](#)
[Design Support](#)
[Support](#)

Design Support

Search for a topic

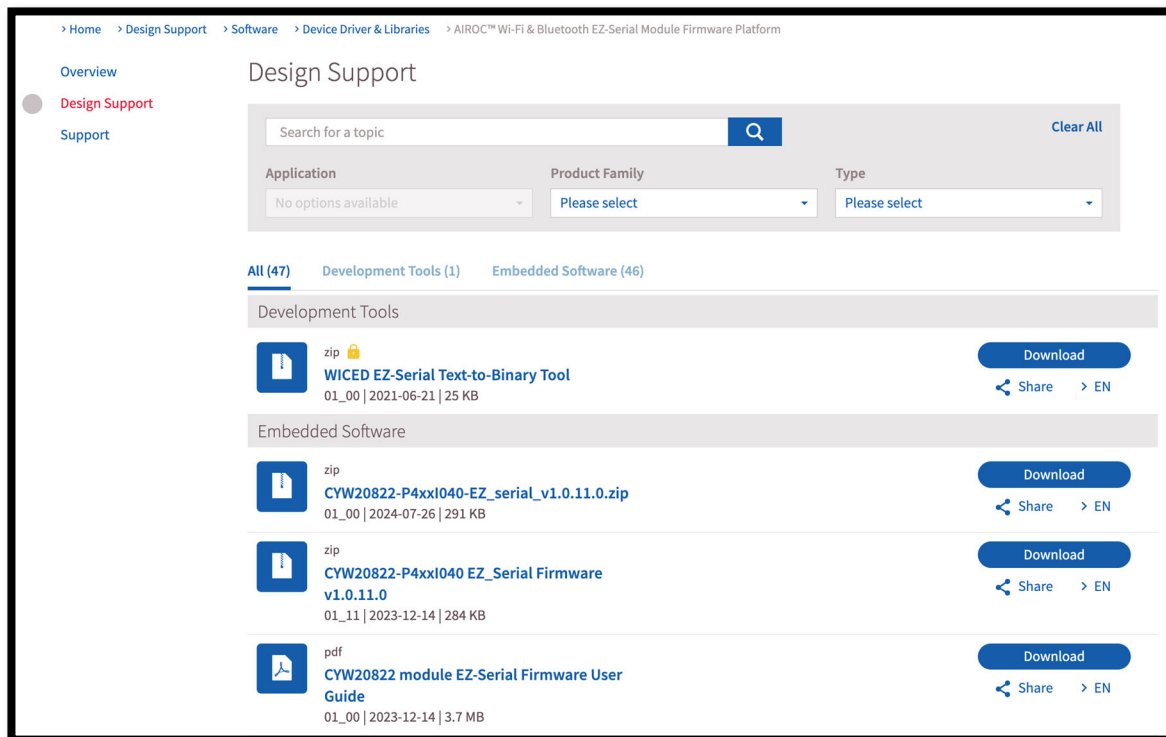
Application: Product Family: Type:

[All \(2\)](#) [Embedded Software \(2\)](#)

Embedded Software

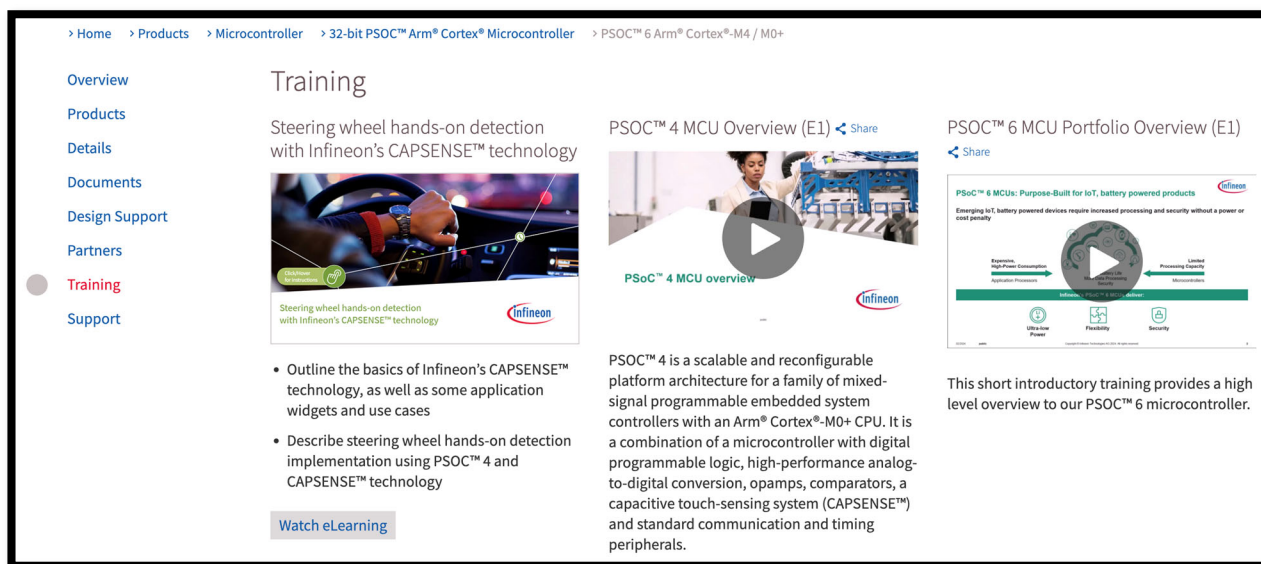
 zip Infineon-cypress-fmac-v5.10.9-2021_1020.zip-Software-v01_00-EN 01_00 2021-12-08 8.5 MB	<input type="button" value="Download"/> <input type="button" value="Share"/> > EN
 zip Infineon-cypress-fmac-v5.4.18-2021_0527.zip-Software-v01_00-EN 01_00 2021-12-08 7.6 MB	<input type="button" value="Download"/> <input type="button" value="Share"/> > EN

(AIROC™ Wi-Fi & Bluetooth Linux and Android Drivers.)



(AIROC™ Wi-Fi & Bluetooth EZ-Serial Module Firmware Platform.)

141. In addition, on its website, Infineon provides training materials and help from Infineon support engineers:



(Exemplary Training Materials for PSoC 6.)

Home > Products > Wireless Connectivity > AIROC™ Wi-Fi and Wi-Fi+Bluetooth® Combo

Overview
Products
Documents
Videos
Training
Support

Training

Introduction to Wi-Fi/Bluetooth® (E1) [Share](#)

AIROC™ Wi-Fi/Bluetooth® Portfolio Overview (E1) [Share](#)

Automotive Wi-Fi and Bluetooth® Combo (E1) [Share](#)

Wi-Fi/Bluetooth® Markets, customers and competitors
Sivaram Thirukumar (QSD ICW R&D)
March 2024

All you wanted to know about Wi-Fi & Bluetooth® but were too shy to ask | A lot of people use Wi-Fi and Bluetooth® regularly in their daily lives among various devices. Check out this training to learn more about the wireless technologies Wi-Fi and Bluetooth®.

This introductory training covers the Infineon AIROC™ Wi-Fi/ Bluetooth® portfolio and how the products map to different applications.

This introductory training covers Infineon's automotive Wi-Fi and Bluetooth® Combo products portfolio offerings and provides use cases for Wi-Fi and Bluetooth®.

(Exemplary Training Materials for AIROC products.)

— Ask the PSoC™ 6 Community

Infineon Developer Community

Ask Answer Share
Connect Collaborate

Infineon Developer Community is available 24x7 for you to connect and network with engineers across the globe. Get help from Infineon support engineers, and expert members to solve your design challenges anytime, from anywhere, in any topic, and in your preferred language.

[Ask the Community](#)

(Link to the Infineon Developer Community for the PSoC 6.)

— Ask the AIROC™ Wi-Fi + Bluetooth Combos Community

Infineon Developer Community

Ask Answer Share
Connect Collaborate

Infineon Developer Community is available 24x7 for you to connect and network with engineers across the globe. Get help from Infineon support engineers, and expert members to solve your design challenges anytime, from anywhere, in any topic, and in your preferred language.

[Ask the Community](#)

(Link to the Infineon Developer Community for the AIROC products.)

142. Infineon knew or should have known of the '306 Patent and its infringement through a series of correspondence and meetings expressly notifying Infineon of the '306 Patent

and Infineon's infringement thereof. As a result, Infineon knew or should have known it infringed one or more claims of the '306 Patent, including at least Claim 16, at least as early as November 3, 2017, when MOSAID began discussions with Cypress regarding the '306 Patent and provided actual notice of infringement.

143. Alternatively, Infineon knew or should have known it infringed one or more claims of the '306 Patent, including at least Claim 16, at least as early as April 17, 2018.

144. At a minimum, Infineon knew or should have known of the '306 Patent and its infringement no later than April 2020, when Infineon acquired Cypress and continued discussions with MOSAID regarding the '306 Patent and Infineon's infringement thereof.

145. Despite this knowledge, Infineon continued to induce third parties, including Infineon's customers and/or end-users of the Accused Products, including at least the PSoC 6 and AIROC products, to infringe one or more claims of the '306 Patent, including at least Claim 16, with the specific intent to cause infringement. Infineon knew or should have known that those acts would induce actual infringement by third parties, including Infineon's customers and/or end-users of the Accused Products, including at least the PSoC 6 and AIROC products, of one or more of the claims of the '306 Patent, including at least Claim 16.

146. Therefore, Infineon has induced infringement by others of one or more of the claims of the '306 Patent, including at least Claim 16, with the specific intent to induce acts that constitute infringement of the '306 Patent and with knowledge that such acts infringed one or more claims of the '306 Patent, including at least Claim 16.

147. In violation of 35 U.S.C. § 271(c), Infineon has infringed one or more of the claims of the '306 Patent, including at least Claim 16, indirectly by contributing to infringement by third parties, including for example Infineon's customers and/or end-users of the Accused Products,

including at least the PSoC 6 and AIROC products, in this District and elsewhere in the United States. Direct infringement by Infineon's customers occurred at least by the use of the Accused Products, including at least the PSoC 6 and AIROC products, including use of consumer products incorporating them.

148. On information and belief, Infineon made and sold hardware, firmware, and/or software components (*e.g.*, processors and/or software drivers) especially made or especially adapted to practice the invention claimed in the '306 Patent, including at least Claim 16. For example, as explained above, Infineon made and sold PSoC and MCU hardware, including the PSoC 6 and AIROC product architectures, and accompanying software, firmware, and driver code. On information and belief, such hardware, firmware, and/or software components (i) are a material part of the invention and (ii) are not staple articles or commodities of commerce suitable for substantial non-infringing use at least because they are specifically designed to perform the claimed functionality. Any other use of such hardware, firmware, and/or software would be unusual, far-fetched, illusory, impractical, occasional, aberrant, or experimental.

149. Therefore, Infineon has contributed to the infringement by others of one or more of the claims of the '306 Patent, including at least Claim 16.

150. Infineon's infringement of one or more claims of the '306 Patent, including at least Claim 16, was willful.

151. As explained above, Infineon had actual knowledge of the '306 Patent and its infringement thereof at least as early as November 3, 2017, when MOSAID began discussions with Cypress regarding the '306 Patent, and in any event no later than April 2020, when Infineon acquired Cypress and continued discussions with MOSAID.

152. As explained above, on several occasions beginning at least as early as November 3, 2017, MOSAID notified Infineon (including Cypress) of the '306 Patent and Infineon's infringement thereof, including infringement of at least Claim 16.

153. Despite knowing of the '306 Patent at least as early as November 3, 2017 (and no later than April 2020), Infineon did not cease its infringing activities. Infineon continued to infringe one or more claims of the '306 Patent, including at least Claim 16, in disregard of MOSAID's patent rights. As a result, Infineon deliberately and intentionally infringed the '306 Patent, and continued to do so, after receiving express and actual knowledge of both the '306 Patent and its infringement thereof.

154. Therefore, Infineon's infringement of the '306 Patent, including at least Claim 16, was willful, wanton, malicious, in bad faith, deliberate, consciously wrongful, flagrant, or characteristic of a pirate, entitling MOSAID to increased damages pursuant to 35 U.S.C. § 284 and to attorneys' fees and costs incurred in prosecuting this action pursuant to 35 U.S.C. § 285.

COUNT III: INFRINGEMENT OF THE '448 PATENT

155. MOSAID incorporates by reference and realleges paragraphs 1 through 154 as if specifically set forth herein.

156. In violation of 35 U.S.C. § 271(a), Infineon is and has been directly infringing one or more of the claims of the '448 Patent, including at least Claim 4, either literally and/or under the doctrine of equivalents, by making, using, selling, and/or offering for sale in the United States, and/or importing into the United States, without authority, at least one of the Accused Products.

157. Claim 4 of the '448 Patent recites:

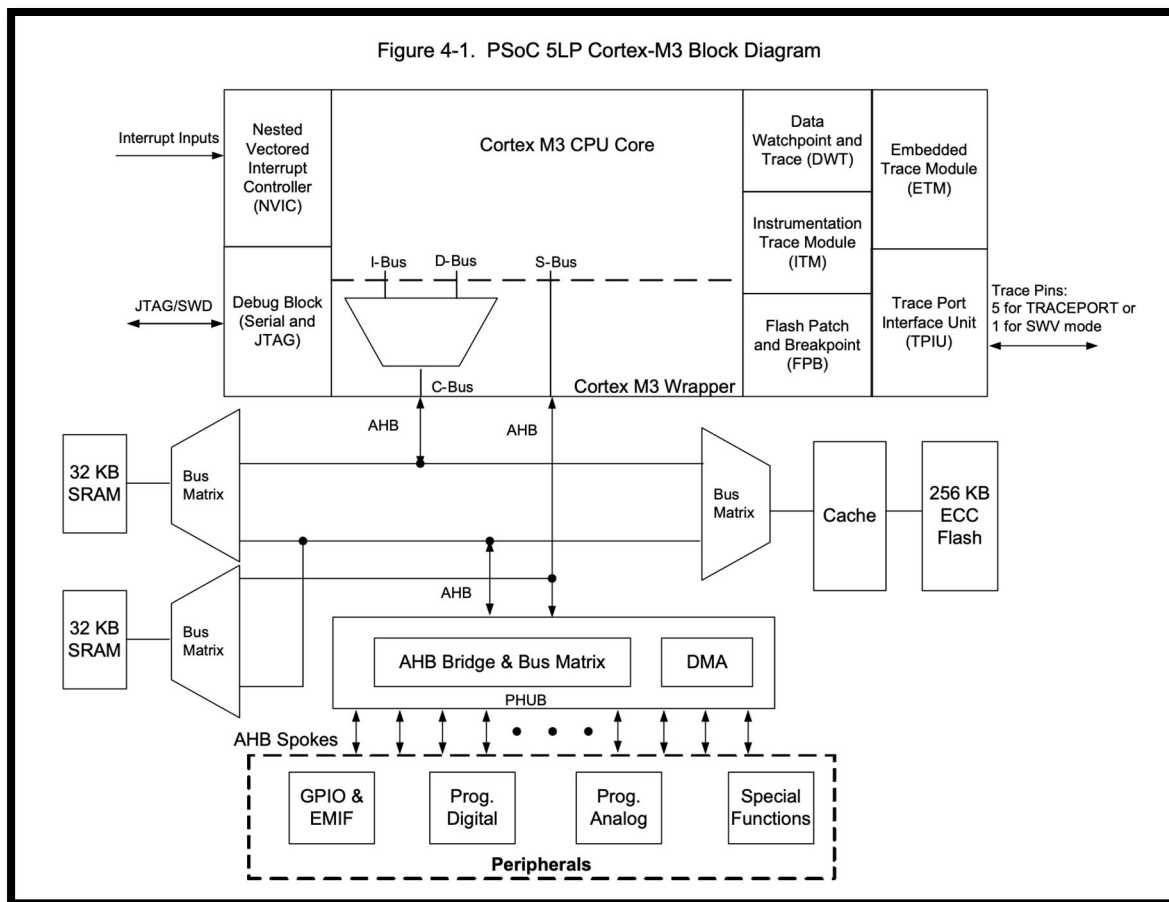
4. A multiplexor circuit, comprising:
a plurality of data connections;

first stage logic configured to receive a first data word from one of the connections and to transmit the first data word received based on a set of select signals; and

second stage logic configured to receive the first data word from the first stage logic and to select a selected data word between the first data word and a second data word received from another of the plurality of data connections based upon a set of select signals, the second stage logic configured to transmit the selected data word to a third stage logic and to increase a signal strength of the first data word when the first data word is transmitted, if the first data word is selected.

158. The Accused Products, including at least the PSoC 5LP family of processors, practice each element of Claim 4 of the '448 Patent.

159. The Accused Products include a multiplexor circuit comprising a plurality of data connections. For example, at least the PSoC 5LP includes circuitry allowing for signal multiplexing and logic operations. As shown below, the circuitry includes data connections to and from various system components including at least an ARM Cortex-M3 CPU, a peripheral hub (“PHUB”) with a direct memory access (“DMA”) controller, and memory components, among others.



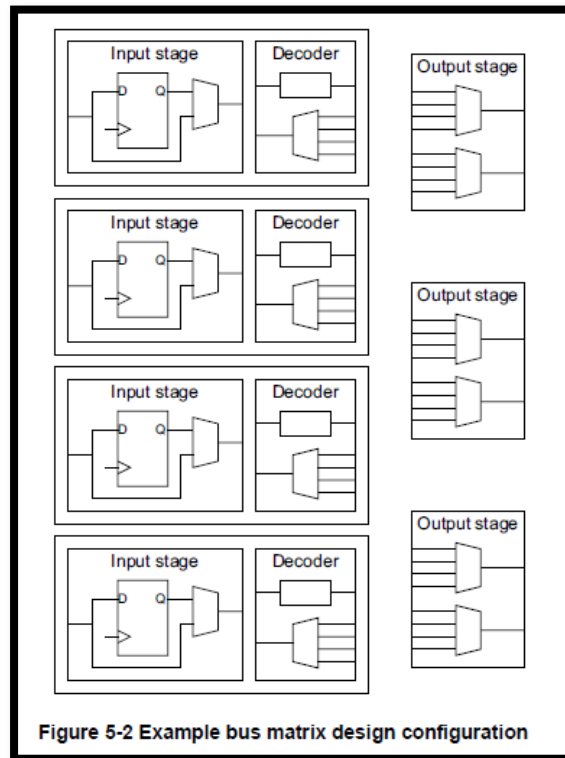
(PSoC 5LP Architecture Technical Reference Manual, at 34.)

160. The Accused Products include a first stage logic configured to receive a first data word from one of the connections. For example, at least the PSoC 5LP includes multiplexer logic (e.g., a bus matrix) configured to receive a first data word via at least the D-Bus and/or S-Bus of the Cortex-M3 CPU.

161. Further, the first stage logic is configured to transmit the first data word received based on a set of select signals. For example, the bus matrix logic of the CPU can decode the access request and assert a select signal for transmission to the appropriate destination. If the bus matrix determines that the CPU wants to access the SRAM or a certain region of SRAM, for example, the bus matrix transmits the first data word to the SRAM/region. In addition, for simultaneous

accesses, the bus matrix logic can determine which access takes priority and transmit the data word accordingly.

162. The Accused Products include a second stage logic configured to receive the first data word from the first stage logic. For example, at least the PSoC 5LP contains additional bus matrix logic for various slave components, including at least SRAM and Flash memory modules. As shown in Figure 4-1, above, the PSoC 5LP uses bus interfaces based on ARM's Advanced High Performance Bus ("AHB") specification. As shown in the figure below, the AHB bus matrix module generally consists of an input stage, a decoder stage, and an output stage. On information and belief, the bus matrices of the Accused Products, including at least the PSoC 5LP, follow this configuration. In the example of an SRAM access by the CPU, the input/decoder stage of the SRAM bus matrix logic will receive the first data word from the first stage logic of the CPU.



(ARM Cortex-M System Design Kit Technical Reference Manual, at 5-6.)

163. Further, the second stage logic is configured to select a selected data word between the first data word and a second data word received from another of the plurality of data connections based upon a set of select signals. For example, the input/decoder stage of the SRAM bus matrix can receive a second data word from another data connection, such as the DMA controller, because the CPU and DMA share bandwidth as dual bus masters. The decoder stage will generate and assert a select signal for the appropriate master-slave connection. In addition, when the CPU and DMA controller attempt to access the same region of SRAM simultaneously, the bus matrix logic can determine which access to permit first. For example, the priority decoder gives a higher priority to the CPU for certain SRAM accesses but gives a higher priority to the DMA controller for other SRAM accesses.

164. In addition, the second stage logic is configured to transmit the selected data word to a third stage logic and to increase a signal strength of the first data word when the first data word is transmitted, if the first data word is selected. For example, the selected data is transmitted to a third logic stage as it is further transmitted to the SRAM. On information and belief, when the first data word is selected and transmitted, there is an increase to its signal strength.

165. During license discussions, MOSAID provided Infineon with exemplary claim charts explaining in detail Infineon's infringement of the '448 Patent.

166. In violation of 35 U.S.C. § 271(b), Infineon is and has been infringing one or more of the claims of the '448 Patent, including at least Claim 4, indirectly by inducing infringement by third parties, including for example Infineon's customers and/or end-users of the Accused Products, in this District and elsewhere in the United States. For example, on information and belief, at least the PSoC 5LP has been and is being used by Infineon's customers and/or end-users in consumer products, including wearables, fitness products, and mobile devices. Direct

infringement by Infineon's customers and/or end-users occurs at least by the use of the Accused Products, including at least the PSoC 5LP, including use of consumer products incorporating them.

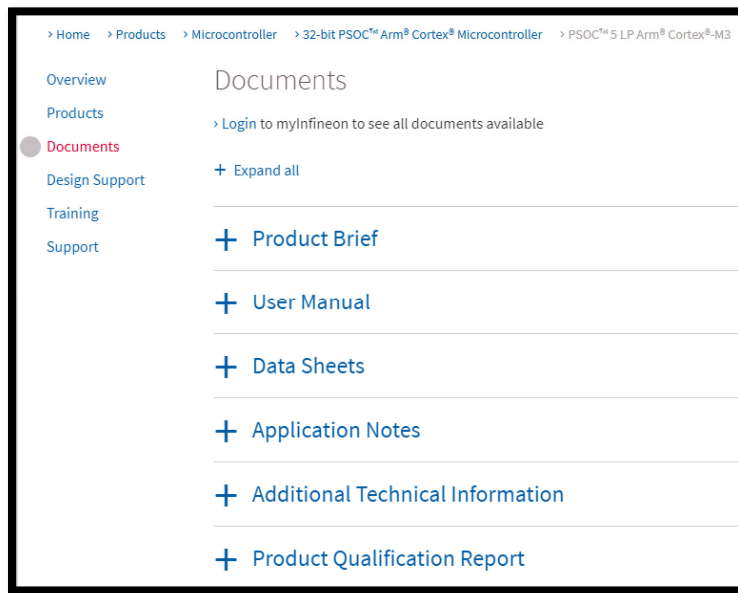
167. On information and belief, Infineon supplies hardware, firmware, and/or software, including software drivers, that are especially made or especially adapted to practice the inventions claimed in the '448 Patent, including at least Claim 4, to induce third parties, including for example Infineon's customers and/or end-users of at least one of the Accused Products, to use the Accused Products in a manner that would infringe one or more claims of the '448 Patent, including at least Claim 4.

168. On information and belief, Infineon supplies hardware, firmware, and/or software that are especially made or especially adapted to practice the inventions claimed in the '448 Patent, including at least Claim 4, to induce third parties, including for example Infineon's customers and/or end-users of the Accused Products, including at least the PSoC 5LP, to use such products in a manner that would infringe one or more of the claims of the '448 Patent, including at least Claim 4.

169. On information and belief, Infineon markets and advertises the Accused Products, including at least the PSoC 5LP, including on its website, to induce third parties, including Infineon's customers and/or end-users, to use the products in a manner that would infringe one or more claims of the '448 Patent, including at least Claim 4. *See, e.g.,* <https://www.infineon.com/cms/en/product/microcontroller/32-bit-psoc-arm-cortex-microcontroller/32-bit-psoc-5-lp-arm-cortex-m3>.

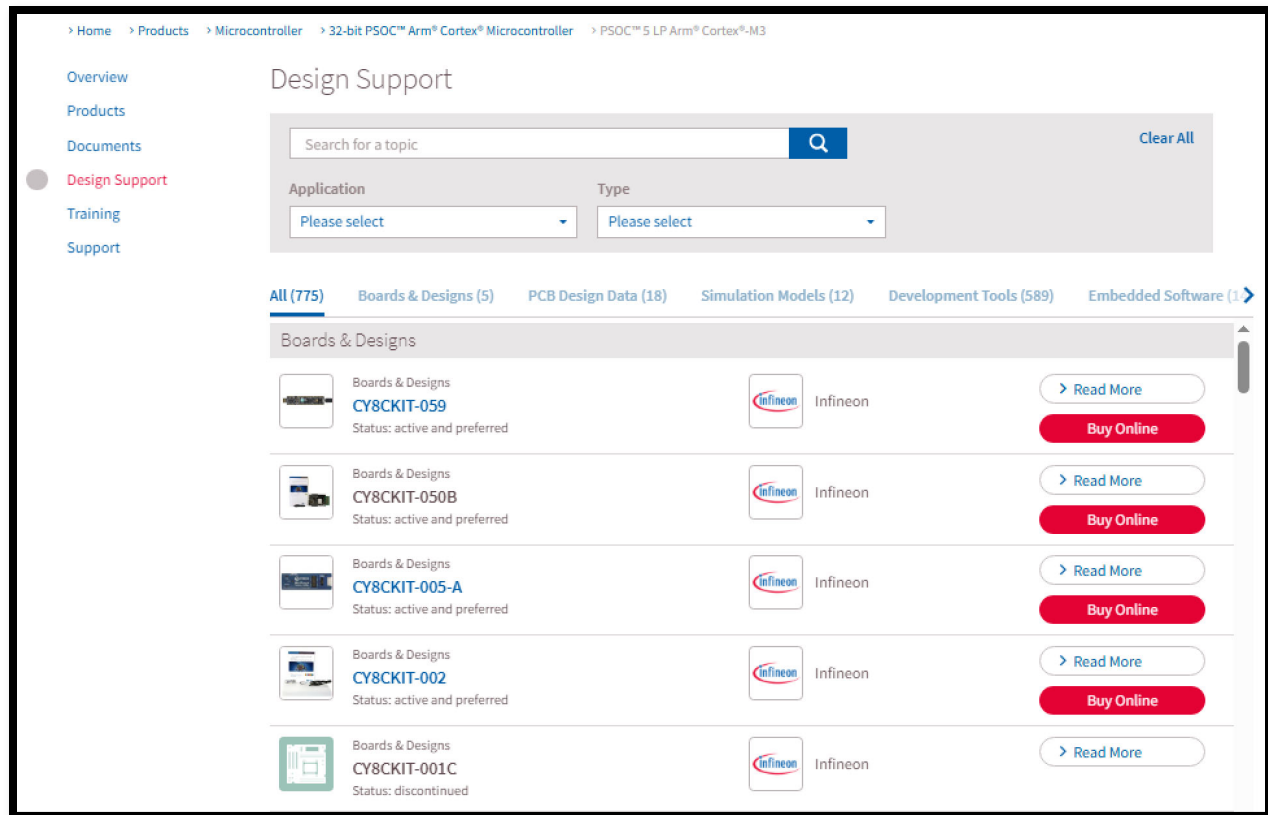
170. On information and belief, Infineon furnishes instructive materials, technical support, and information concerning the operation and use of the Accused Products, including at least the PSoC 5LP, to induce third parties, including Infineon's customers and/or end-users, to

use the products in a manner that would infringe one or more claims of the '448 Patent, including at least Claim 4. For example, on its website, Infineon furnishes at least product briefs, user manuals, data sheets, application notes, additional product and technical information (such as architecture technical reference manuals), and product qualification reports:



(Exemplary categories of instructive materials provided for the PSoC 5LP, with additional materials provided for each product subcategory under their respective web subpages.)

171. On its website, Infineon also furnishes development tools and design support materials for the Accused Products, including at least the PSoC 5LP:



(Exemplary design support materials for PSoC 5LP.)

172. Further, Infineon provides software, including code examples and drivers:

> Home > Design Support > Software > Code Examples > PSOC™ 3/4/5 Code Examples for PSOC™ Creator

PSOC™ 3/4/5 Code Examples

Overview
Downloads
Design Support
Support

The code examples linked in the table below are compatible with PSOC™ Creator 3.0 SP2. To access the latest code examples, follow the path File -> Example Projects in PSOC™ Creator. To build with a different version of PSOC™ Creator, first update the project components in Creator by following the path Project -> Update Components.

- + Code examples compatible with PSOC™ Creator 3.0 SP2
- + Code Examples integrated with Application Notes
- + Code Examples integrated with Development Kits (CY8CKIT-050 PSOC™ 5LP Development Kit)

> Home > Design Support > Software > Code Examples > PSOC™ 3/4/5 Code Examples for PSOC™ Creator

Code Examples integrated with Development Kits (CY8CKIT-050 PSOC™ 5LP Development Kit)

The following Code Examples are integrated with CY8CKIT-050 PSOC™ 5LP Development Kit.

Title	Part #	Description
VoltageDisplay_SAR_ADC	PSOC™ 5LP	This example code measures an analog voltage controlled by the potentiometer. The code uses the internal SAR ADC configured for a 12-bit operation; the ADC range is 0 to V _{dda} . The results are displayed on the character LCD module.
VoltageDisplay_DelSigADC	PSOC™ 5LP	This example code measures a simple analog voltage controlled by the potentiometer. The code uses the internal Del-Sig ADC configured for a 20-bit operation; the ADC range is 0 to V _{dda} . The voltage measurement resolution is in microvolts. The results are displayed on the character LCD module.
IntensityLED	PSOC™ 5LP	This example code uses a pulse-width modulator (PWM) to illuminate an LED. When the pulse width of the PWM varies, the LED brightness changes. By continuously varying the pulse width of the PWM, the example code makes an LED go from low brightness to a high brightness and back.
LowPowerDemo	PSOC™ 5LP	This code example demonstrates the low-power functionality of PSOC™ 5LP. The project implements an RTC based code, which goes to sleep and wakes up on the basis of switch inputs. The RTC uses an accurate 32-kHz clock generated using the external crystal provided on the board. When there is a key press, the device is put to sleep while the RTC is kept active.
CAPSENSE™	PSOC™ 5LP	This code example provides a platform to build CAPSENSE™-based projects using PSOC™ 5LP. The example uses two CAPSENSE™ buttons and one five-element slider provided on the board. Each capacitive sensor on the board is scanned using the Infineon CSD algorithm. The buttons are pretuned in the example code to take care of factors such as board parasitic.
ADC_DAC	PSOC™ 5LP	This project demonstrates sine wave generation by using an 8-bit DAC and DMA. The sine wave period is based on the

The firmware reads the voltage output by the board potentiometer and displays the raw counts on the LCD. An 8-bit DAC outputs a table generated sine wave to an LED using DMA at a frequency proportional to the ADC count, current value of the ADC value of the potentiometer.

(Infineon PSOC 3/4/5 Code Examples, including code examples integrated with PSoC 5LP Development Kit.)

4 Code Examples

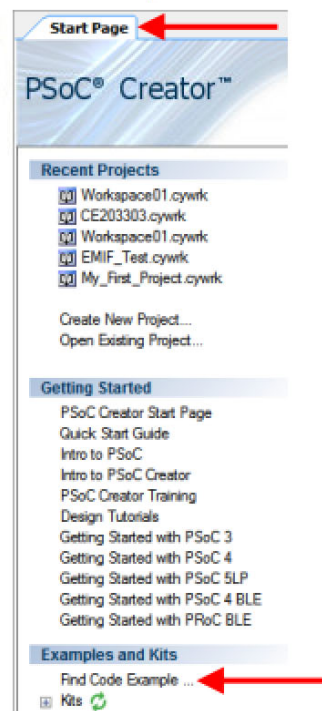
PSoC Creator includes a large number of code example projects. These code examples are available from the PSoC Creator Start Page, as Figure 2 shows.

Code examples can speed up your design process by starting you off with a complete design, instead of a blank page. The code examples also show how PSoC Creator Components are used in various applications. Code examples and datasheets are included, as Figure 3 shows.

In the Find Code Example dialog shown in Figure 3, you have several options:

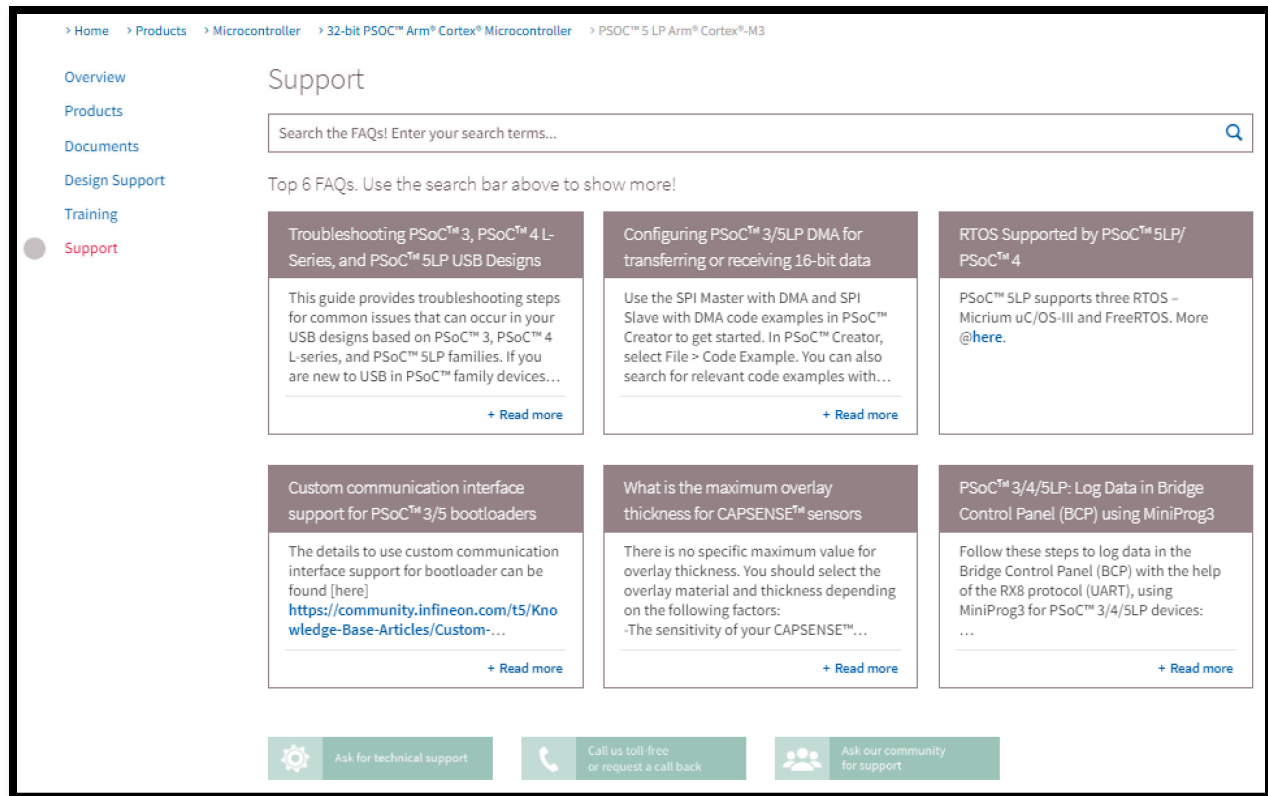
- Filter for examples based on architecture or device family, such as, PSoC 3, PSoC 4 or PSoC 5LP; category; or keyword
- Select from the menu of examples offered based on the Filter Options
- Review the datasheet for the selection (on the **Documentation** tab)
- Review the code example for the selection. You can copy and paste code from this window to your project, which can help speed up code development, or
- Create a new project (and a new workspace if needed) based on the selection. This can speed up your design process by starting you off with a complete, basic design. You can then adapt that design to your application.

Figure 2. Code Examples in PSoC Creator

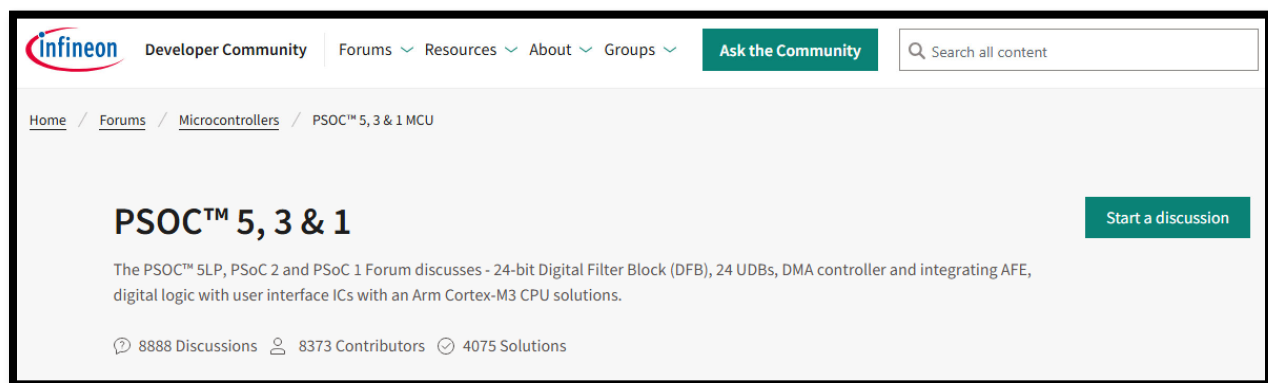


(Getting Started with PSoC 5LP, at 4.)

173. In addition, on its website, Infineon provides training materials and help from Infineon support engineers:



(Exemplary Training Materials for PSoC 5LP.)



(Infineon Developer Community for PSoC 5LP.)

174. Infineon knew or should have known of the '448 Patent and its infringement through a series of correspondence and meetings expressly notifying Infineon of the '448 Patent and Infineon's infringement thereof. As a result, Infineon knew or should have known it infringed one or more claims of the '448 Patent, including at least Claim 4, at least as early as November 3,

2017, when MOSAID began discussions with Cypress regarding the '448 Patent and provided actual notice of infringement.

175. Alternatively, Infineon knew or should have known of the '448 Patent and its infringement no later than April 2020, when Infineon acquired Cypress and continued discussions with MOSAID regarding the '448 Patent and Infineon's infringement thereof.

176. At a minimum, both Infineon Defendants have had actual knowledge of the '448 Patent, and their infringement thereof, at least as of the date of filing of this Complaint.

177. Despite this knowledge, Infineon has continued to induce third parties, including Infineon's customers and/or end-users of at least one of the Accused Products, including at least the PSoC 5LP, to infringe one or more claims of the '448 Patent, including at least Claim 4, with the specific intent to cause infringement. Infineon knew or should have known that those acts would induce actual infringement by third parties, including Infineon's customers and/or end-users of the Accused Products, including at least the PSoC 5LP, of one or more of the claims of the '448 Patent, including at least Claim 4.

178. Therefore, Infineon has induced infringement by others of one or more of the claims of the '448 Patent, including at least Claim 4, with the specific intent to induce acts that constitute infringement of the '448 Patent and with knowledge that such acts infringe one or more claims of the '448 Patent, including at least Claim 4.

179. In violation of 35 U.S.C. § 271(c), Infineon is and has been infringing one or more of the claims of the '448 Patent, including at least Claim 4, indirectly by contributing to infringement by third parties, including for example Infineon's customers and/or end-users of the Accused Products, including at least the PSoC 5LP, in this District and elsewhere in the United States. Direct infringement by Infineon's customers and/or end-users occurs at least by the use of

the Accused Products, including at least the PSoC 5LP, including use of consumer products incorporating them.

180. On information and belief, Infineon made and sold hardware, firmware, and/or software components (*e.g.*, processors and/or software drivers) especially made or especially adapted to practice the invention claimed in the '448 Patent, including at least Claim 4. For example, as explained above, Infineon made and sold PSoC hardware, including the PSoC 5LP architecture, and accompanying code. On information and belief, such hardware, firmware, and/or software components (i) are a material part of the invention and (ii) are not staple articles or commodities of commerce suitable for substantial non-infringing use at least because they are specifically designed to perform the claimed functionality. Any other use of such hardware, firmware, and/or software would be unusual, far-fetched, illusory, impractical, occasional, aberrant, or experimental.

181. Therefore, Infineon has contributed to the infringement by others of one or more of the claims of the '448 Patent, including at least Claim 4.

182. Infineon's infringement of one or more claims of the '448 Patent, including at least Claim 4, has been, and continues to be, willful.

183. As explained above, Infineon had actual knowledge of the '448 Patent and its infringement thereof at least as early as November 3, 2017, when MOSAID began discussions with Cypress regarding the '448 Patent, and in any event no later than April 2020, when Infineon acquired Cypress and continued discussions with MOSAID.

184. As explained above, on several occasions beginning at least as early as November 3, 2017, MOSAID notified Infineon (including Cypress) of the '448 Patent and Infineon's infringement thereof.

185. Despite knowing of the '448 Patent at least as early as November 3, 2017 (and no later than April 2020), Infineon did not cease its infringing activities. Infineon has continued to infringe one or more claims of the '448 Patent, including at least Claim 4, in disregard of MOSAID's patent rights. As a result, Infineon deliberately and intentionally infringed the '448 Patent, and continues to do so, after receiving express and actual knowledge of both the '448 Patent and its infringement thereof.

186. Therefore, Infineon's infringement of the '448 Patent, including at least Claim 4, has been and continues to be willful, wanton, malicious, in bad faith, deliberate, consciously wrongful, flagrant, or characteristic of a pirate, entitling MOSAID to increased damages pursuant to 35 U.S.C. § 284 and to attorneys' fees and costs incurred in prosecuting this action pursuant to 35 U.S.C. § 285.

DAMAGES

187. Defendants' acts of infringement have caused damages to MOSAID, and MOSAID is entitled to recover the damages it has sustained as a result of Defendants' wrongful acts in an amount to be determined at trial.

188. MOSAID is entitled to, and now seeks to, recover damages in an amount not less than the maximum amount permitted by law caused by Defendants' acts of infringement.

189. As a result of Defendants' acts of infringement, MOSAID has suffered actual and consequential damages. To the fullest extent permitted by law, MOSAID seeks recovery of damages in an amount to compensate for Defendants' infringement. MOSAID further seeks any other damages to which MOSAID would be entitled to in law or in equity.

INJUNCTIVE RELIEF

190. Defendants' acts of infringement have caused—and unless restrained and enjoined, Defendants' acts of infringement will continue to cause—irreparable injury and damage to

MOSAID for which MOSAID has no adequate remedy at law. Unless enjoined by this Court, Defendants will continue to infringe the claims of the Asserted Patents.

ATTORNEYS' FEES

191. MOSAID is entitled to recover reasonable and necessary attorneys' fees under applicable law.

DEMAND FOR JURY TRIAL

192. Pursuant to Rule 38 of the Federal Rules of Civil Procedure, MOSAID demands a trial by jury on all issues so triable.

PRAYER FOR RELIEF

WHEREFORE, MOSAID prays for judgment and requests that the Court find in its favor and against Defendants. MOSAID respectfully requests that the Court enter preliminary and final orders, declarations, and judgments against Defendants as are necessary to provide MOSAID with the following relief:

- a. A judgment that Defendants have infringed and/or are infringing one or more claims of the Asserted Patents, literally or under the doctrine of equivalents, as alleged above;
- b. A judgment that Defendants have infringed and/or are infringing one or more claims of the Asserted Patents, directly, as alleged above;
- c. A judgment that Defendants have infringed and/or are infringing one or more claims of the Asserted Patents, indirectly, as alleged above;
- d. A judgment that Defendants' infringement of the claims of the Asserted Patents has been willful;
- e. An award for all damages and costs arising out of Defendants' infringement, to adequately compensate MOSAID for Defendants' infringement of the Asserted Patents,

but in no event less than a reasonable royalty, including supplemental damages for any continuing post-verdict infringement up until entry of the final judgment, with an accounting, as needed;

f. Pre-judgment and post-judgment interest, jointly and severally, in an amount according to proof;

g. Treble damages based on Defendants' willful infringement;

h. An accounting of damages and any future compensation due to MOSAID for Defendants' infringement (past, present, or future) not specifically accounted for in a damages award (or other relief), and/or permanent injunctive relief;

i. A judgment that this case is exceptional and an award of reasonable attorneys' fees as provided by 35 U.S.C. § 285 and enhanced damages as provided by 35 U.S.C. § 284;

j. An award of costs of suit;

k. The entry of an order enjoining and restraining Defendants and their parents, affiliates, subsidiaries, officers, agents, servants, employees, attorneys, successors, and assigns and all persons in active concert or participation therewith, from making, importing, using, offering for sale, selling, or causing to be sold any product falling within the scope of any claim of the Asserted Patents, or otherwise infringing or inducing infringement of any claim of the Asserted Patents; and

l. All further relief in law or in equity as the Court may deem just and proper.

Dated: March 10, 2025

Respectfully submitted,

/s/ Jamie H. McDole

Max Ciccarelli (SBN 00787242)

CICCARELLI LAW FIRM

100 N 6th Street, Suite 503

Waco, Texas 76701

Telephone: 214-444-8869

Email: max@ciccarellilawfirm.com

Jamie H. McDole (SBN 24082049)

Lead Counsel

Phillip B. Philbin (SBN 15909020)

Michael D. Karson (SBN 24090198)

Miranda Y. Jones (SBN 24065519)

David W. Higer (SBN 24127850)

David A. Neal (SBN 24144868)

Cody M. Carter (SBN 24131091)

Matthew L. Vitale (SBN 24137699)

WINSTEAD PC

2728 N. Harwood Street, Suite 500

Dallas, Texas 75201

Tel.: (214) 745-5400

Fax: (214) 745-5390

Email: jmcdole@winstead.com

pphilbin@winstead.com

mkarson@winstead.com

mjones@winstead.com

dhiger@winstead.com

dneal@winstead.com

ccarter@winstead.com

mvitale@winstead.com

Nadia E. Haghighatian (SBN 24087652)

Austin C. Teng (SBN 24093247)

Daniel S. Shuminer (SBN 24124688)

WINSTEAD PC

600 W. 5th Street, Suite 900

Austin, Texas 78701

Tel.: (512) 370-2800

Fax: (512) 370-2850

Email: nhaghighatian@winstead.com

ateng@winstead.com

dshuminer@winstead.com

Attorneys for Plaintiff

MOSAID Technologies Inc.

CIVIL COVER SHEET

The JS 44 civil cover sheet and the information contained herein neither replace nor supplement the filing and service of pleadings or other papers as required by law, except as provided by local rules of court. This form, approved by the Judicial Conference of the United States in September 1974, is required for the use of the Clerk of Court for the purpose of initiating the civil docket sheet. (SEE INSTRUCTIONS ON NEXT PAGE OF THIS FORM.)

I. (a) PLAINTIFFS

MOSAID TECHNOLOGES INC.

(b) County of Residence of First Listed Plaintiff Ottawa, ON Canada
(EXCEPT IN U.S. PLAINTIFF CASES)

(c) Attorneys (Firm Name, Address, and Telephone Number)

SEE ATTACHED

DEFENDANTS

INFINEON TECHNOLOGIES AG
INFINEON TECHNOLOGIES AMERICAS CORP.

County of Residence of First Listed Defendant Germany
(IN U.S. PLAINTIFF CASES ONLY)

NOTE: IN LAND CONDEMNATION CASES, USE THE LOCATION OF THE TRACT OF LAND INVOLVED.

Attorneys (If Known)

II. BASIS OF JURISDICTION (Place an "X" in One Box Only)

- ☐ 1 U.S. Government Plaintiff ☒ 3 Federal Question (U.S. Government Not a Party)
- ☐ 2 U.S. Government Defendant ☐ 4 Diversity (Indicate Citizenship of Parties in Item III)

III. CITIZENSHIP OF PRINCIPAL PARTIES (Place an "X" in One Box for Plaintiff and One Box for Defendant)

- | | PTF | DEF | | PTF | DEF |
|---|----------------------------|----------------------------|---|----------------------------|----------------------------|
| Citizen of This State | ☐ 1 | ☐ 1 | Incorporated or Principal Place of Business In This State | ☐ 4 | ☐ 4 |
| Citizen of Another State | ☐ 2 | ☐ 2 | Incorporated and Principal Place of Business In Another State | ☐ 5 | ☐ 5 |
| Citizen or Subject of a Foreign Country | ☐ 3 | ☐ 3 | Foreign Nation | ☐ 6 | ☐ 6 |

IV. NATURE OF SUIT (Place an "X" in One Box Only)

Click here for: [Nature of Suit Code Descriptions.](#)

CONTRACT	TORTS	FORFEITURE/PENALTY	BANKRUPTCY	OTHER STATUTES
☐ 110 Insurance ☐ 120 Marine ☐ 130 Miller Act ☐ 140 Negotiable Instrument ☐ 150 Recovery of Overpayment & Enforcement of Judgment ☐ 151 Medicare Act ☐ 152 Recovery of Defaulted Student Loans (Excludes Veterans) ☐ 153 Recovery of Overpayment of Veteran's Benefits ☐ 160 Stockholders' Suits ☐ 190 Other Contract ☐ 195 Contract Product Liability ☐ 196 Franchise	PERSONAL INJURY ☐ 310 Airplane ☐ 315 Airplane Product Liability ☐ 320 Assault, Libel & Slander ☐ 330 Federal Employers' Liability ☐ 340 Marine ☐ 345 Marine Product Liability ☐ 350 Motor Vehicle ☐ 355 Motor Vehicle Product Liability ☐ 360 Other Personal Injury ☐ 362 Personal Injury - Medical Malpractice PERSONAL INJURY ☐ 365 Personal Injury - Product Liability ☐ 367 Health Care/ Pharmaceutical Personal Injury Product Liability ☐ 368 Asbestos Personal Injury Product Liability PERSONAL PROPERTY ☐ 370 Other Fraud ☐ 371 Truth in Lending ☐ 380 Other Personal Property Damage ☐ 385 Property Damage Product Liability	☐ 625 Drug Related Seizure of Property 21 USC 881 ☐ 690 Other LABOR ☐ 710 Fair Labor Standards Act ☐ 720 Labor/Management Relations ☐ 740 Railway Labor Act ☐ 751 Family and Medical Leave Act ☐ 790 Other Labor Litigation ☐ 791 Employee Retirement Income Security Act IMMIGRATION ☐ 462 Naturalization Application ☐ 465 Other Immigration Actions	☐ 422 Appeal 28 USC 158 ☐ 423 Withdrawal 28 USC 157 INTELLECTUAL PROPERTY RIGHTS ☐ 820 Copyrights ☒ 830 Patent ☐ 835 Patent - Abbreviated New Drug Application ☐ 840 Trademark ☐ 880 Defend Trade Secrets Act of 2016 SOCIAL SECURITY ☐ 861 HIA (1395ff) ☐ 862 Black Lung (923) ☐ 863 DIWC/DIWW (405(g)) ☐ 864 SSID Title XVI ☐ 865 RSI (405(g)) FEDERAL TAX SUITS ☐ 870 Taxes (U.S. Plaintiff or Defendant) ☐ 871 IRS—Third Party 26 USC 7609	☐ 375 False Claims Act ☐ 376 Qui Tam (31 USC 3729(a)) ☐ 400 State Reapportionment ☐ 410 Antitrust ☐ 430 Banks and Banking ☐ 450 Commerce ☐ 460 Deportation ☐ 470 Racketeer Influenced and Corrupt Organizations ☐ 480 Consumer Credit (15 USC 1681 or 1692) ☐ 485 Telephone Consumer Protection Act ☐ 490 Cable/Sat TV ☐ 850 Securities/Commodities/Exchange ☐ 890 Other Statutory Actions ☐ 891 Agricultural Acts ☐ 893 Environmental Matters ☐ 895 Freedom of Information Act ☐ 896 Arbitration ☐ 899 Administrative Procedure Act/Review or Appeal of Agency Decision ☐ 950 Constitutionality of State Statutes
REAL PROPERTY ☐ 210 Land Condemnation ☐ 220 Foreclosure ☐ 230 Rent Lease & Ejectment ☐ 240 Torts to Land ☐ 245 Tort Product Liability ☐ 290 All Other Real Property	CIVIL RIGHTS ☐ 440 Other Civil Rights ☐ 441 Voting ☐ 442 Employment ☐ 443 Housing/Accommodations ☐ 445 Amer. w/Disabilities - Employment ☐ 446 Amer. w/Disabilities - Other ☐ 448 Education PRISONER PETITIONS Habeas Corpus: ☐ 463 Alien Detainee ☐ 510 Motions to Vacate Sentence ☐ 530 General ☐ 535 Death Penalty Other: ☐ 540 Mandamus & Other ☐ 550 Civil Rights ☐ 555 Prison Condition ☐ 560 Civil Detainee - Conditions of Confinement			

V. ORIGIN (Place an "X" in One Box Only)

- ☒ 1 Original Proceeding ☐ 2 Removed from State Court ☐ 3 Remanded from Appellate Court ☐ 4 Reinstated or Reopened ☐ 5 Transferred from Another District (specify) ☐ 6 Multidistrict Litigation - Transfer ☐ 8 Multidistrict Litigation - Direct File

VI. CAUSE OF ACTION

Cite the U.S. Civil Statute under which you are filing (Do not cite jurisdictional statutes unless diversity):
35 U.S.C. § 271
Brief description of cause:
Patent Infringement

VII. REQUESTED IN COMPLAINT:

☐ CHECK IF THIS IS A CLASS ACTION UNDER RULE 23, F.R.Cv.P. DEMAND \$ CHECK YES only if demanded in complaint:
JURY DEMAND: ☒ Yes ☐ No

VIII. RELATED CASE(S) IF ANY

(See instructions):

JUDGE DOCKET NUMBER

DATE Mar 10, 2025 SIGNATURE OF ATTORNEY OF RECORD /s/ Jamie H. McDole

FOR OFFICE USE ONLY

RECEIPT # AMOUNT APPLYING IFP JUDGE MOSAID Technologies Inc., Exhibit 2004

ATTACHMENT

Max Ciccarelli

CICCARELLI LAW FIRM

100 N 6th Street, Suite 503

Waco, Texas 76701

Telephone: 214-444-8869

Email: max@ciccarellilawfirm.com

Jamie H. McDole

Lead Counsel

Phillip B. Philbin

Michael D. Karson

Miranda Y. Jones

David W. Higer

David A. Neal

Cody M. Carter

Matthew L. Vitale

WINSTEAD PC

2728 N. Harwood Street, Suite 500

Dallas, Texas 75201

Tel.: (214) 745-5400

Fax: (214) 745-5390

Email: jmcdole@winstead.com

pphilbin@winstead.com

mkarson@winstead.com

mjones@winstead.com

dhiger@winstead.com

dneal@winstead.com

ccarter@winstead.com

mvitale@winstead.com

Nadia E. Haghighatian

Austin C. Teng

Daniel S. Shuminer

WINSTEAD PC

600 W. 5th Street, Suite 900

Austin, Texas 78701

Tel.: (512) 370-2800

Fax: (512) 370-2850

Email: nhaghighatian@winstead.com

ateng@winstead.com

dshuminer@winstead.com

Attorneys for Plaintiff

MOSAID Technologies Inc.

EXHIBIT A



US007111179B1

(12) **United States Patent**
Girson et al.

(10) **Patent No.:** **US 7,111,179 B1**
(45) **Date of Patent:** **Sep. 19, 2006**

(54) **METHOD AND APPARATUS FOR
OPTIMIZING PERFORMANCE AND
BATTERY LIFE OF ELECTRONIC DEVICES
BASED ON SYSTEM AND APPLICATION
PARAMETERS**

(75) Inventors: **Andrew Girson**, Gaithersburg, MD
(US); **Boris Donskoy**, Herndon, VA
(US); **Nathan Tennes**, N. Chevy
Chase, MD (US)

(73) Assignee: **In-Hand Electronics, Inc.**, Rockville,
MD (US)

(*) Notice: Subject to any disclaimer, the term of this
patent is extended or adjusted under 35
U.S.C. 154(b) by 101 days.

(21) Appl. No.: **10/268,914**

(22) Filed: **Oct. 11, 2002**

Related U.S. Application Data

(60) Provisional application No. 60/333,535, filed on Oct.
11, 2001.

(51) **Int. Cl.**
G06F 1/32 (2006.01)

(52) **U.S. Cl.** **713/300; 713/320; 320/130**

(58) **Field of Classification Search** 320/135,
320/155, 130-132, 156; 324/426; 702/62,
702/64; 713/321-323, 300, 320

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

3,922,526 A	11/1975	Cochran	713/321
4,137,563 A	1/1979	Tsunoda	713/601
4,670,837 A	6/1987	Sheets	713/501
4,823,292 A	4/1989	Hillion	713/321
4,851,987 A	7/1989	Day	713/601
4,893,271 A	1/1990	Davis et al.	713/501
4,979,507 A	12/1990	Heinz et al.	607/28
5,086,387 A	2/1992	Arroyo et al.	713/501

5,142,684 A	8/1992	Perry et al.	713/320
5,189,647 A	2/1993	Suzuki et al.	368/10
5,218,704 A	6/1993	Watts, Jr. et al.	713/322
5,230,056 A	7/1993	Hoshina	713/340
5,230,074 A	7/1993	Canova, Jr. et al.	714/14
5,237,692 A	8/1993	Raasch et al.	710/267
5,239,652 A	8/1993	Seibert et al.	713/323
5,245,629 A	9/1993	Hall	455/522
5,307,003 A	4/1994	Fairbanks et al.	323/222
5,325,048 A *	6/1994	Longini	324/74
5,339,445 A	8/1994	Gasztanyi	713/324
5,369,771 A	11/1994	Gettel	713/322
5,383,137 A	1/1995	Burch	703/23
5,386,577 A	1/1995	Zenda	713/340
5,416,726 A	5/1995	Garcia-Duarte et al.	713/300
5,452,277 A	9/1995	Bajorek et al.	369/53.18
5,475,847 A	12/1995	Ikeda	713/322
5,493,684 A	2/1996	Gephardt et al.	713/322
5,493,685 A	2/1996	Zenda	713/340
5,504,776 A	4/1996	Yamaura et al.	375/141
5,504,908 A	4/1996	Ikeda	713/300

(Continued)

FOREIGN PATENT DOCUMENTS

EP 0363567 4/1990

(Continued)

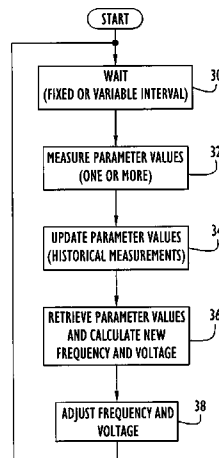
Primary Examiner—Donald Sparks

(74) *Attorney, Agent, or Firm*—Edell, Shapiro & Finnan,
LLC

(57) **ABSTRACT**

An electronic device (e.g., computer system, etc.) employing dynamic power management of the present invention adjusts power consumption in accordance with an analysis of parameters and events occurring over one or more time-periods. Preferably, the electronic device monitors microprocessor, operating system, peripheral and/or device-level events and adjusts run-time parameters, such as microprocessor clock frequency and voltage, to reduce power consumption with minimal perceived degradation in performance.

42 Claims, 6 Drawing Sheets



US 7,111,179 B1

Page 2

U.S. PATENT DOCUMENTS

5,524,249 A	6/1996	Suboh	713/322	5,991,883 A	11/1999	Atkinson	713/300
5,546,568 A	8/1996	Bland et al.	713/501	5,996,084 A	11/1999	Watts	713/323
5,585,745 A	12/1996	Simmons et al.	326/93	6,006,336 A	12/1999	Watts, Jr. et al.	713/322
5,585,750 A		Noguchi et al.	327/113	6,044,282 A	3/2000	Hlasny	455/573
5,600,229 A	2/1997	Oh	320/162	6,073,244 A	6/2000	Iwazaki	713/322
5,619,707 A	4/1997	Suboh	713/322	6,078,209 A	6/2000	Linoff	327/513
5,623,647 A	4/1997	Maitra	713/501	6,079,022 A	6/2000	Young	713/300
5,625,826 A	4/1997	Atkinson	713/322	6,079,025 A	6/2000	Fung	713/323
5,628,001 A	5/1997	Cepuran	713/501	6,112,309 A	8/2000	Inoue et al.	713/501
5,664,203 A	9/1997	Hong et al.	713/323	6,122,234 A	9/2000	Fujitani et al.	369/47.4
5,675,808 A	10/1997	Gulick et al.	713/322	6,131,166 A	10/2000	Wong-Insley	713/300
5,680,626 A	10/1997	Chu et al.	713/322	6,138,232 A	10/2000	Shiell et al.	712/244
5,682,273 A	10/1997	Hetzler	360/75	6,158,012 A *	12/2000	Watts, Jr.	713/322
5,691,948 A	11/1997	Sakabe	365/227	6,163,721 A	12/2000	Thompson	607/2
5,692,197 A	11/1997	Narad et al.	713/323	6,173,409 B1	1/2001	Watts, Jr. et al.	713/322
5,699,336 A	12/1997	Maeda et al.	369/53.18	6,201,371 B1	3/2001	Kawabe et al.	320/121
5,708,821 A	1/1998	Takikita	713/310	6,266,776 B1	7/2001	Sakai	713/300
5,715,467 A	2/1998	Jirgal	713/300	6,269,043 B1	7/2001	Batcher	365/227
5,721,935 A	2/1998	DeSchepper et al.	713/323	6,282,661 B1	8/2001	Nicol	713/300
5,734,711 A	3/1998	Kleffner	379/323	6,298,448 B1	10/2001	Shaffer et al.	713/322
5,748,103 A	5/1998	Flach et al.	340/870.07	6,304,761 B1	10/2001	Tsunehiro	340/7.32
5,752,011 A	5/1998	Thomas et al.	713/501	6,304,978 B1	10/2001	Horigan et al.	713/322
5,752,045 A	5/1998	Chen	713/322	6,311,287 B1	10/2001	Dischler et al.	713/601
5,754,867 A	5/1998	Walker	713/322	6,340,874 B1	1/2002	Vladimir	318/471
5,758,172 A	5/1998	Seo	713/300	6,356,615 B1	3/2002	Coon et al.	377/16
5,777,914 A	7/1998	Larsson et al.	708/322	6,363,491 B1	3/2002	Endo	713/310
5,781,783 A	7/1998	Gunther et al.	713/320	6,367,023 B1	4/2002	Kling et al.	713/340
5,794,146 A	8/1998	Sevcik et al.	455/434	6,407,689 B1	6/2002	Bazarjani et al.	341/143
5,799,198 A	8/1998	Fung	713/323	6,425,068 B1	7/2002	Vorbach et al.	712/18
5,815,693 A	9/1998	McDermott et al.	713/501	6,427,211 B1	7/2002	Watts, Jr.	713/320
5,825,156 A	10/1998	Patillon et al.	702/63	2001/0001880 A1	5/2001	Watts, Jr.	713/322
5,825,674 A	10/1998	Jackson	713/321	2001/0005892 A1	6/2001	Watts, Jr. et al.	713/300
5,832,280 A	11/1998	Swanberg	713/300	2001/0044909 A1	11/2001	Oh et al.	713/600
5,861,006 A	1/1999	Kroll	607/5	2002/0004915 A1	1/2002	Fung	713/320
5,864,702 A	1/1999	Walsh et al.	713/320	2002/0029353 A1	3/2002	Hwang	713/322
5,892,959 A	4/1999	Fung	713/323	2002/0032535 A1 *	3/2002	Alexander et al.	702/64
5,913,068 A	6/1999	Matoba	713/322	2003/0020421 A1 *	1/2003	Vu et al.	318/442
5,926,640 A	7/1999	Mason et al.	713/320	2003/0158677 A1 *	8/2003	Swarztrauber et al.	702/62
5,930,516 A	7/1999	Watts, Jr. et al.	713/322				
5,948,105 A	9/1999	Skurnik et al.	713/323				
5,953,536 A	9/1999	Nowlin, Jr.	713/323				
5,974,556 A	10/1999	Jackson et al.	713/322				
5,983,357 A	11/1999	Sun	713/324				

FOREIGN PATENT DOCUMENTS

EP	0712064	5/1996
JP	11338744	12/1999
WO	WO/0153921	7/2001

* cited by examiner

U.S. Patent

Sep. 19, 2006

Sheet 1 of 6

US 7,111,179 B1

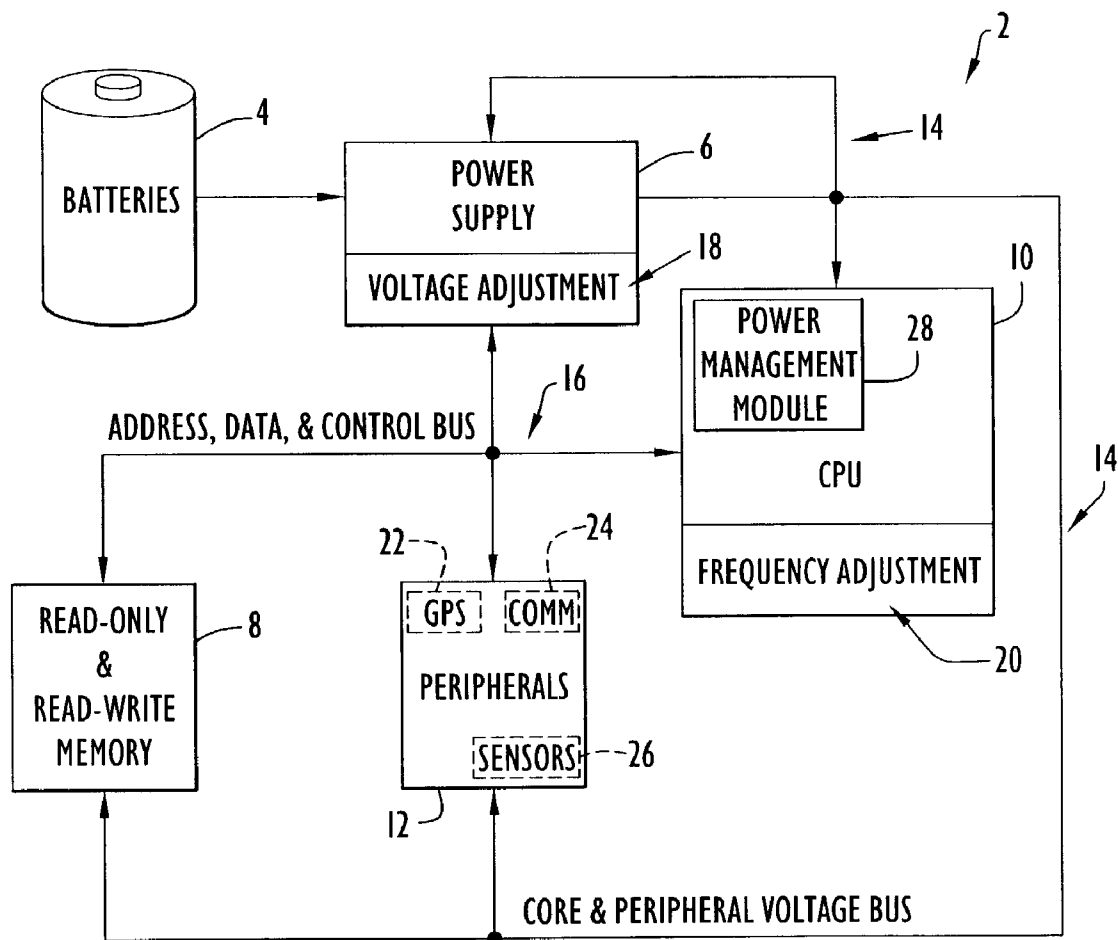


FIG. 1

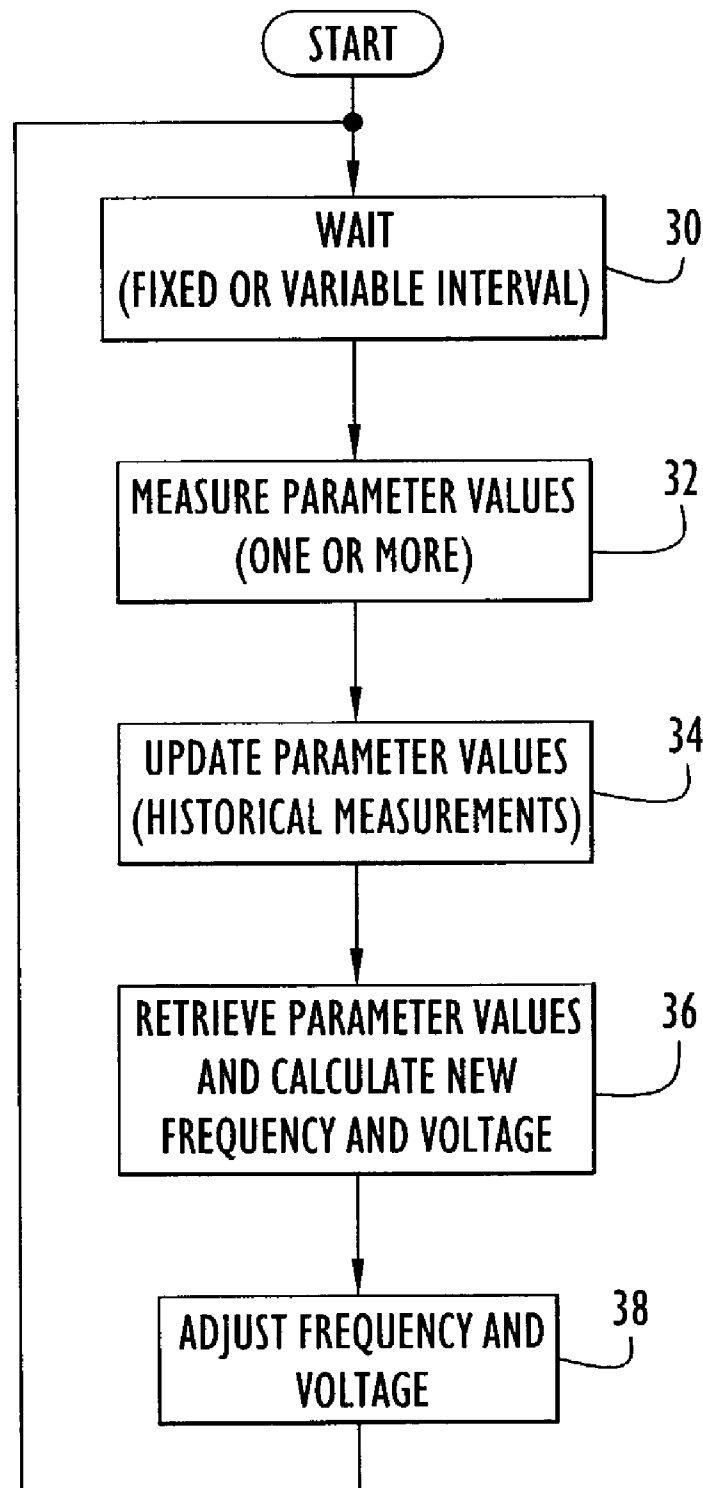


FIG.2

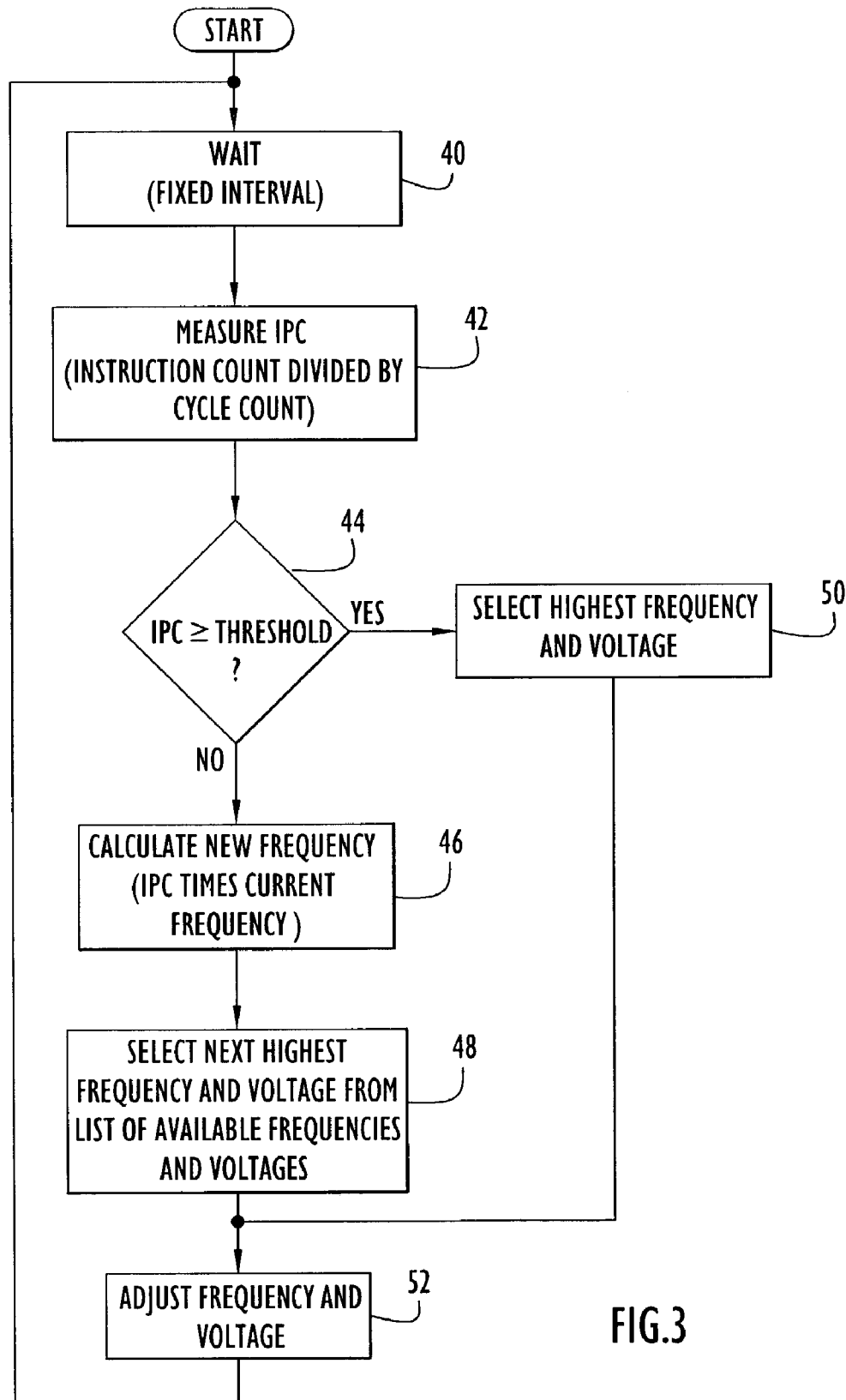


FIG.3

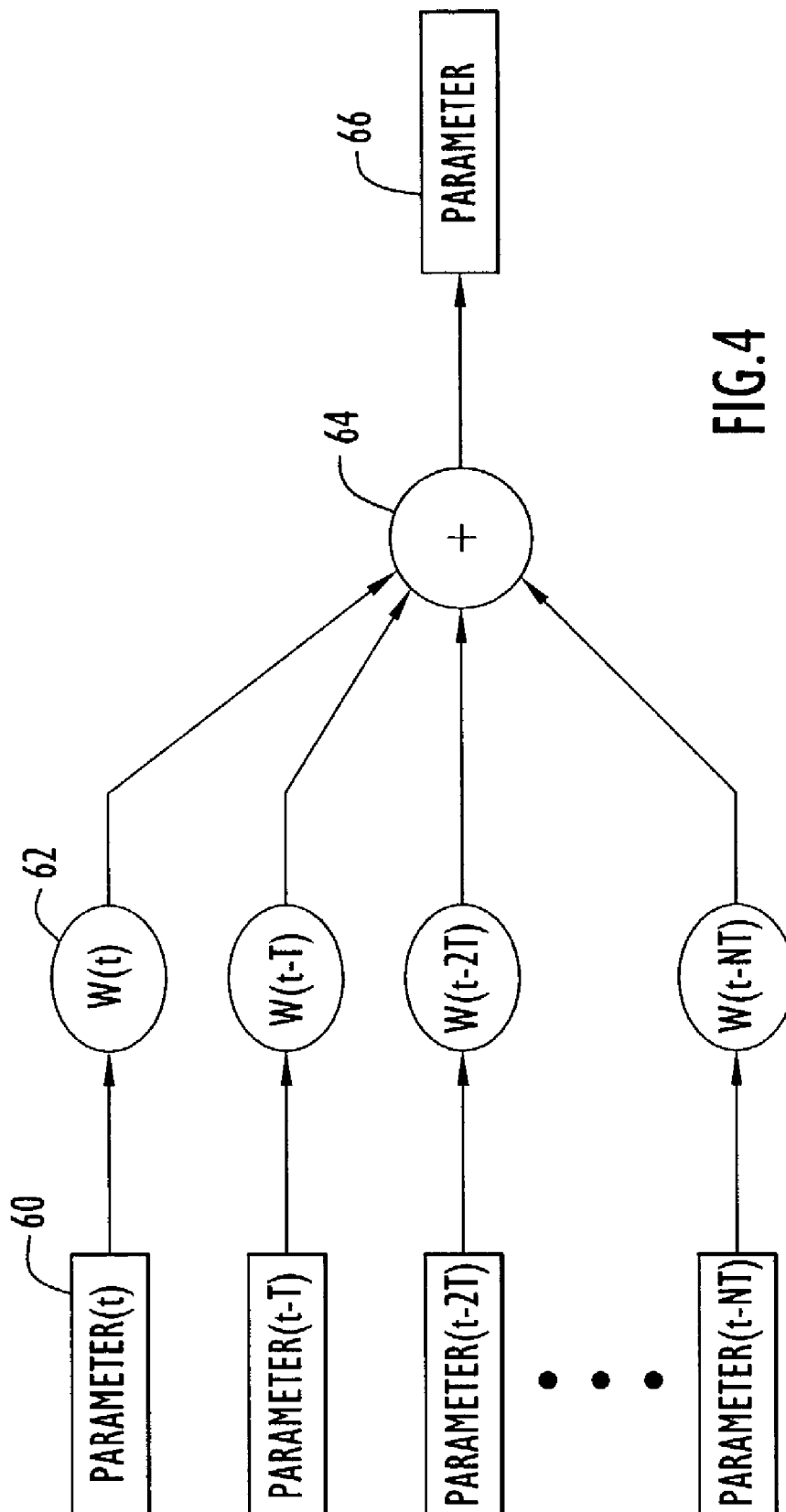


FIG.4

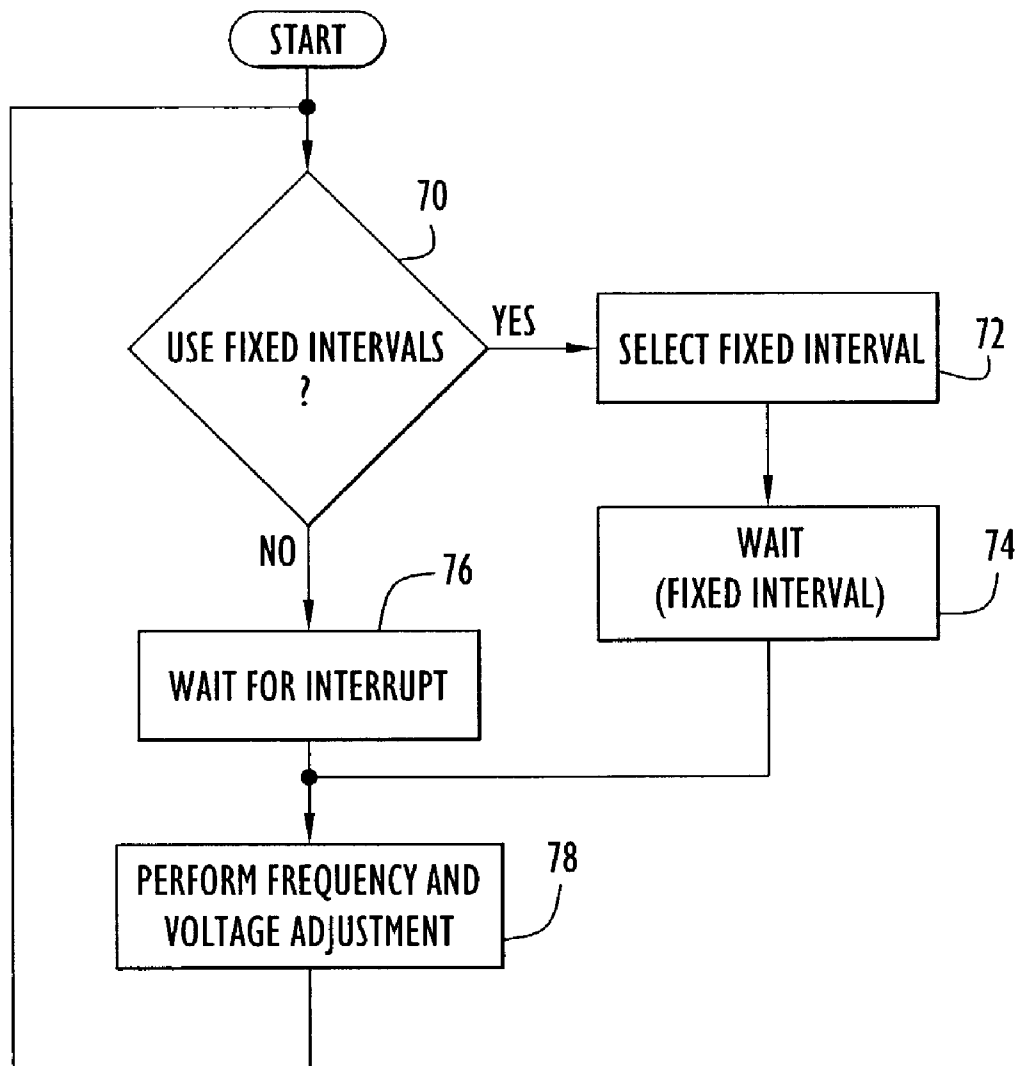


FIG.5

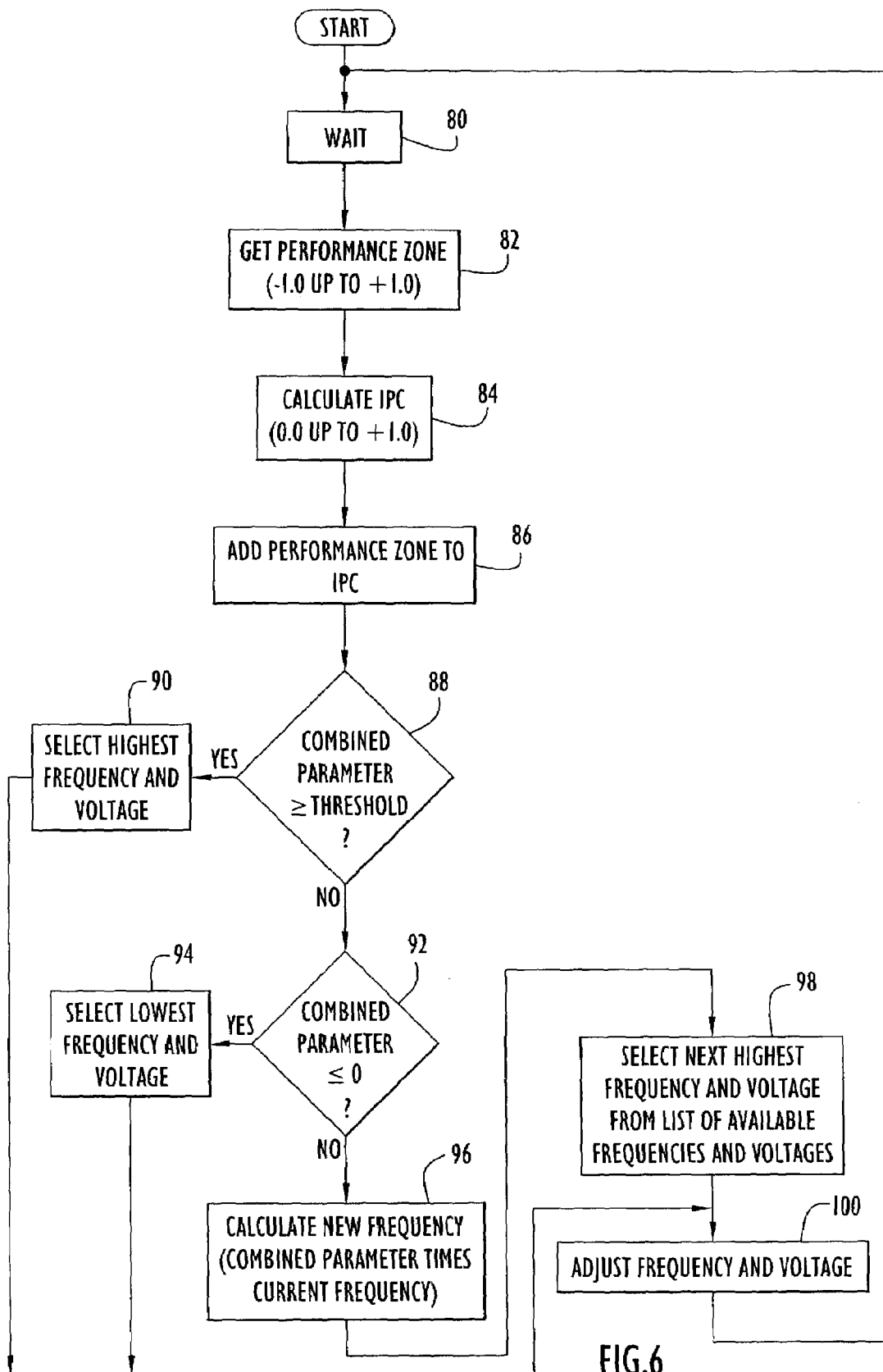


FIG. 6

US 7,111,179 B1

1

**METHOD AND APPARATUS FOR
OPTIMIZING PERFORMANCE AND
BATTERY LIFE OF ELECTRONIC DEVICES
BASED ON SYSTEM AND APPLICATION
PARAMETERS**

**CROSS-REFERENCE TO RELATED
APPLICATIONS**

This application claims priority to U.S. Provisional Patent Application Ser. No. 60/333,535, entitled "Performance and Battery Life Optimization Based on System and Application Parameters" and filed Oct. 11, 2001, the disclosure of which is incorporated herein by reference in its entirety.

BACKGROUND OF THE INVENTION

1. Field of the Invention

The present invention relates to processing systems or devices and, more particularly, to adjustment of power used during operation of those devices.

2. Discussion of the Related Art

Portable and autonomously powered electronic devices (e.g., computer systems, telephones, PDAs, etc.) are becoming more and more sophisticated, while at the same time, becoming smaller. These devices, whether powered by batteries, fuel cells, solar power, or some other power source, are capable of performing functions that have been traditionally relegated to large desktop and server computer systems.

While the technology for portable electronic devices has progressed rapidly, improvements in battery capacity and energy density have not been as rapid. As a result, a gap has developed between the processing capabilities of portable electronic devices and the battery life of those devices. This gap has constrained the "autonomy" of portable electronic devices, reducing the amount of time between battery recharge or replacement, and thus limiting the true portability of these devices. Often, designers of portable electronic devices must delicately balance battery parameters such as size, weight, and cost against the portability and operational characteristics of the device.

In an attempt to overcome these problems, computer electronics manufacturers have incorporated many features into their hardware and software designs to extend battery life. A feature that has been introduced is frequency and voltage adjustment for microprocessors. Various companies, such as Intel, IBM, Texas Instruments, Motorola, and Transmeta, have introduced microprocessors that include the run-time capability to increase or decrease the frequency and voltage at which the microprocessor core operates. Since power consumption is directly proportional to the clock frequency of the microprocessor and to the square of the corresponding applied voltage, reductions in both parameters can have a cubic effect on overall power consumption of the device. However, reducing clock frequency may also reduce instruction-processing performance.

The related art provides several practices of understanding performance appropriately and adjusting performance in attempt to overcome this problem. For example, when activity of a computer system is below a threshold for a certain period of time, hardware components of the computer system are shut off or deactivated to conserve power. By way of example, a display backlight may be turned off or the computer system itself may be disabled in response to a user not providing any input to the computer system for several minutes. Alternatively, the microprocessor core may

2

be placed in a non-processing low-power state with the clock stopped. Examples of such approaches are disclosed in U.S. Pat. Nos. 5,386,577, 5,493,685, 5,983,357 and/or 5,991,883, the disclosures of which are incorporated herein by reference in their entireties.

Further, the clock frequency of a microprocessor may be reduced when activity of a computer system is below a threshold for a certain period of time, and subsequently increased in response to increased computer system activity. Examples of such approaches are disclosed in U.S. Pat. Nos. 5,369,771, 5,628,001, 5,752,011 and/or 5,983,357, the disclosures of which are incorporated herein by reference in their entireties.

In addition, the types of activities that a computer system performs may be monitored to determine whether clock frequency and voltage are to be reduced or increased. Examples of such approaches are disclosed in U.S. Pat. Nos. 5,504,908, 5,625,826, 5,892,959, 5,996,084 and/or 6,427,211, the disclosures of which are incorporated herein by reference in their entireties.

The related art suffers from several disadvantages. In particular, the related art techniques provide some straightforward capabilities to reduce power consumption through various optimizations; however, the power reduction achieved is limited. Further, the related art techniques monitor individual or a few basic parameters to reduce power, thereby providing a simplistic approach that may enter a power conserving state at an inopportune time and consequently adversely affect performance.

The present invention overcomes the aforementioned problems and provides several advantages. In particular, the present invention significantly reduces power consumption (effectively increasing battery life by a factor of two or more), while not appreciably affecting electronic device performance. These power consumption reductions are achieved through a combination of optimizations, aggregated together and analyzed over plural time-periods. The optimizations are based on very low-level system parameters related to the microprocessor hardware and operating system software, and high-level parameters related to the application and end-user operating environment.

**OBJECTS AND SUMMARY OF THE
INVENTION**

Accordingly, it is an object of the present invention to significantly reduce the power consumption of electronic devices, thereby extending device operating life.

It is another object of the present invention to adjust operating frequency and/or voltage of an electronic device in accordance with an analysis of individual or a combination of one or more low-level parameters, such as instructions-per-cycle, thread zones, and/or interrupt density.

Yet another object of the present invention is to adjust operating frequency and/or voltage of an electronic device in accordance with an analysis of individual or a combination of one or more environmental parameters, such as weather conditions, physical location, orientation, motion, time-of-day and/or received wireless radio signal strength.

Still another object of the present invention is to adjust operating frequency and/or voltage of an electronic device in accordance with an analysis of the above-mentioned parameters conducted in one or more of the following manners: analysis of the parameters over plural time-periods (e.g., including current and historical periods); the time-periods being of equal duration (e.g., fixed sampling rate) or of variable duration (e.g., based on factors such as operating

US 7,111,179 B1

3

system task/thread switches or interrupt arrival); and/or analysis of plural parameters simultaneously, with each parameter providing a fixed or adjustable scaling or voting factor in determining if a specified power-saving threshold is achieved.

The aforesaid objects may be achieved individually and/or in combination, and it is not intended that the present invention be construed as requiring two or more of the objects to be combined unless expressly required by the claims attached hereto.

According to the present invention, an electronic device (e.g., computer system, etc.) adjusts power consumption in accordance with an analysis of parameters and events occurring over one or more time-periods. Preferably, the electronic device monitors microprocessor, operating system, peripheral and/or device-level events and adjusts run-time parameters, such as microprocessor clock frequency and voltage, to reduce power consumption with minimal perceived degradation in performance.

The above and still further objects, features and advantages of the present invention will become apparent upon consideration of the following detailed description of specific embodiments thereof, particularly when taken in conjunction with the accompanying drawings wherein like reference numerals in the various figures are utilized to designate like components.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of an exemplary computer system employing the dynamic power management of the present invention.

FIG. 2 is a procedural flow chart illustrating the manner in which dynamic power management is accomplished in accordance with the present invention.

FIG. 3 is a procedural flow chart illustrating the manner in which dynamic power management is achieved based on an instructions-per-cycle (IPC) parameter in accordance with the present invention.

FIG. 4 is a functional block diagram illustrating the manner in which values for a parameter are combined over plural time-periods for power management determinations in accordance with the present invention.

FIG. 5 is a procedural flow chart illustrating the manner in which an interval type is selected for performing parameter measurements in accordance with the present invention.

FIG. 6 is a procedural flowchart illustrating the manner in which plural parameters are combined in a series of stages for power management determinations in accordance with the present invention.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

An exemplary computer system employing the dynamic power management of the present invention is illustrated in FIG. 1. By way of example, the present invention is described with respect to a computer system; however, the present invention may be employed by any electronic device including a processing system or processor to reduce power consumption by that device. Specifically, computer system 2 includes a power source or batteries 4, a power supply 6, memories 8, a processor or CPU 10 and peripherals 12. The power supply receives power from batteries 4 and provides appropriate power signals to the various computer system components (e.g., memories 8, processor 10, peripherals 12, etc.) via a power bus 14. The power supply is typically

4

implemented by a conventional power supply and includes a voltage adjustment module 18 (e.g., in the form of hardware (such as circuitry) and/or software) to control the voltage provided by that supply. Processor 10 is typically implemented by a conventional microprocessor (e.g., Intel StrongARM or the like, etc.) including a frequency adjustment module 20 (e.g., in the form of hardware (such as circuitry) and/or software). The frequency adjustment module controls the processor clock rate or frequency.

Memories 8 include the read-only and read-write memories of the computer system (e.g., ROM, RAM, etc.). These memories are typically implemented by conventional components. Peripherals 12 may include various communications or other devices. By way of example, peripherals 12 may include a Global Positioning System (GPS) receiver 22, a communications device or link 24 (e.g., modem, etc.) and/or sensors 26 (e.g., temperature sensors, humidity sensors, etc.). Receiver 22 and link 24 interface external devices (e.g., GPS system, network, Internet, etc.) to receive parameter or other information, while sensors 26 produce parameter measurements. The parameter information and measurements are utilized for power management determinations as described below. The processor communicates with the power supply, memories and peripherals via an information bus 16. The various computer system components (e.g., memories, power supply, processor, peripherals, buses, etc.) are typically implemented by conventional components and/or devices, where their configurations may vary slightly to accommodate particular characteristics of a device. For example, the clock frequency and voltage adjustment modules may be integrated within the processor and power supply or be implemented as external modules.

The processor includes a power management module 28 to monitor various device and/or external parameters and facilitate adjustment of the processor clock frequency and voltage to conserve power in accordance with the present invention as described below. The power management module is typically implemented as a software module or unit executable by processor 10; however, this module may be implemented as a hardware module or unit (e.g., circuitry) or as any combination of hardware and software, and may be external of or integral with the processor. The power management module provides control signals to adjust the processor clock rate and voltage. Frequency adjustment module 20 adjusts the processor clock rate in accordance with the control signals, while the processor communicates with voltage adjustment module 18 to adjust the power supply voltage in accordance with these signals. The power management module typically accesses peripherals 12 and memories 8 to store and receive parameter information for power management determinations as described below. Parameter information may be stored in memories 8 for several parameter sampling intervals, thereby enabling power management determinations based on current parameter measurements and a history of these measurements as described below. In addition, memories 8 include available frequency and voltage settings for the processor and power supply to facilitate adjustment of the clock frequency and voltage as described below. These settings may be stored in the memories in any desired fashion (e.g., list, stack, queue, array, etc.).

The power management module basically facilitates measurement and storage of parameters within a parameter sampling interval, and examines those parameters to adjust the processor clock rate and voltage accordingly. The manner in which dynamic power management is performed by power management module 28 is illustrated in FIG. 2.

US 7,111,179 B1

5

Initially, the power management module waits for the beginning of a parameter sampling interval at step 30. This interval may be fixed (e.g., occurs at a prescribed interval, such as every one millisecond) or variable (e.g., based on system activity, such as interrupts) as described below. The power management module basically monitors a timer or system activity to determine the occurrence of a fixed or variable interval, respectively. Once a parameter sampling interval commences, the power management module facilitates measurement of one or more parameters during that interval at step 32. The measured parameter values are stored in memories 8 at step 34. Current measured values for a parameter may be stored in a queue or other data structure with prior values for that parameter to provide a parameter history for power management determinations as described below. The stored measurement values are retrieved to determine a new processor clock frequency and voltage at step 36. Basically, the parameter values are utilized to select one of the series of available frequency and voltage settings stored in memories 8 (FIG. 1). These memories typically contain appropriate frequency and corresponding voltage settings for the particular device as described above. The selected settings are applied to power supply 6 and processor 10 at step 38 to adjust the voltage and clock frequency, respectively. Generally, greater power is required to operate the processor at a higher frequency. Accordingly, the power management module typically adjusts the voltage prior to the frequency for a frequency increase, and adjusts the frequency prior to the voltage for a frequency reduction. This prevents the situation where the processor is operating at a high frequency with low voltage. The frequency and voltage values determined in a parameter sampling interval are maintained until the next interval.

Power management module 28 may utilize various parameters to determine appropriate frequency and voltage settings, but preferably determines the settings in accordance with the number of instructions-per-cycle (IPC). Instructions-per-cycle is basically determined by counting the number of instructions executed by the processor and the number of clock frequency cycles that occur during a time interval (e.g., parameter sampling interval, desired time interval, etc.) and dividing the instruction count by the cycle count. The number of cycles and instructions executed are typically determined by and retrieved from the processor. The IPC parameter is indicative of system performance and the type of activities that are occurring as described below, and may further be determined relative to any desired unit time other than a processor clock cycle (e.g., instructions per millisecond or other time interval, etc.).

In modern microprocessors, deep pipelining is used to theoretically allow one machine language instruction to be completed for every microprocessor clock cycle, thereby resulting in an IPC of 1.0. IPC typically falls below 1.0 due to factors that cause a slowdown in the execution of instructions. These factors may include system inactivity (idle time), off-chip accesses to memory for cache-line fills and variables, off-chip accesses to peripherals, and pipeline stalls. IPC may be utilized to determine the effect of a reduction of clock frequency and voltage on system performance. For example, when IPC is near 1.0, the system is typically executing tight instruction loops from within the cache. In this case, clock frequency and voltage should generally be increased to allow the instructions to be executed faster. As IPC falls below 1.0, the system is either idling (e.g., not executing instructions) for portions of the time interval (e.g., parameter sampling interval, etc.) or is performing lots of memory or peripheral accesses, which are

6

typically slow relative to the clock frequency. In these cases, clock frequency and voltage should be reduced because the activities will be completed in the time allotted, but power consumption will be reduced.

The manner in which power management module 28 utilizes the IPC parameter to control clock frequency and voltage is illustrated in FIG. 3. Initially, the power management module typically waits for the beginning of a parameter sampling interval at step 40 as described above. This interval is typically fixed (e.g., occurs at a prescribed time interval, preferably on the order of milliseconds or microseconds). Once the parameter sampling interval occurs, the instruction and cycle count information is retrieved from the processor to determine the IPC at step 42 as described above. When the IPC is near 1.0, the processor is constantly occupied with code execution, thereby enabling an increase in clock frequency to increase performance and potentially reduce future bandwidth requirements. If, however, the IPC is moderately less than 1.0, the processor is not fully utilized (e.g., due to idle time, cache misses, etc.) and a reduction in frequency increases IPC, without degradation of performance (provided that the IPC does not reach 1.0). Accordingly, the IPC parameter is compared to a threshold that is set to be slightly less than 1.0, typically in the range between 0.95–0.99 (e.g., $\text{threshold} = 1.0 - \Delta$, where Δ is predetermined or set by a user). As discussed above, when IPC is 1.0, the processor is operating efficiently and an increased clock frequency should be employed. Generally, the IPC parameter is less than 1.0 since the processor usually performs an operation in a given interval that endures for several cycles. The threshold accounts for these instances and basically adjusts sensitivity of the power management module with respect to entering a high performance mode (e.g., increased clock frequency). Thus, the selection of the delta value sets the power management module sensitivity and essentially enables a trade-off between performance and battery life (e.g., the lower the delta value, the longer the battery life and the lower the performance).

When the IPC is greater than or equal to the threshold as determined at step 44, the IPC parameter indicates efficient processor operation and the frequency and voltage are set to the highest frequency and corresponding voltage setting stored in memories 8 (FIG. 1) at step 50. These settings are subsequently applied at step 52 to adjust the clock frequency and voltage. Otherwise (e.g., IPC is less than the threshold), the new frequency is determined by multiplying the IPC by the current frequency setting at step 46. The next highest frequency (relative to the new frequency) within the stored frequency values is selected and the frequency and corresponding voltage setting are retrieved from the stored values at step 48. The new settings are subsequently applied at step 50 to adjust the clock frequency and voltage. The current IPC value multiplied by the current frequency setting produces a frequency yielding a theoretical IPC of 1.0. The selection of the next highest frequency value setting impedes the IPC from attaining a value of 1.0.

By way of example, a processor includes four frequency settings (e.g., 100, 200, 300, and 400 MHz). If the current frequency is 400 MHz and the measured IPC is 0.60, the new frequency should be the next highest possible setting above 240 MHz (e.g., $0.6 \times 400 = 240$ MHz), or 300 MHz. In this case, assuming the processor maintains the same execution profile for the next parameter sampling interval, the IPC value is 0.80, and the calculation indicates that the frequency should remain at 300 MHz. Eventually, the IPC value may be 0.60, indicating that the frequency should be adjusted

US 7,111,179 B1

7

down to 200 MHz, or alternatively, the IPC may be 1.0, indicating that the frequency is to be set to the maximum of 400 MHz.

The power management module may alternatively employ historical filtering to derive more accurate IPC or other parameter values as illustrated in FIG. 4. In particular, measured values for a parameter 60, such as IPC, are retrieved from memories 8. The parameter values for plural parameter sampling intervals are typically stored in memories 8 in a circular buffer or queue of length N, where N is the number of samples of the parameter to include in the filter and a new parameter value measurement is stored in the position of the least-recent parameter value measurement. The parameter values may be stored for any desired quantity of parameter sampling intervals, where t represents a current time and T represents a parameter sampling interval. Thus, the expression 'parameter (t-NT)' represents a measurement from N prior parameter sampling intervals. The retrieved parameters are each multiplied by a corresponding weight 62 and applied to a summer 64. The summer produces a resulting parameter value 66 that is basically a summation of the weighted parameter values. The resulting value is subsequently utilized as an overall value for the parameter (e.g., IPC) to determine clock frequency and voltage adjustments (e.g., in the manner described above for FIG. 3). Thus, the filtering is generally a weighted average, where the most recent parameter values may be given more weight than older prior values based on the corresponding weights. The effect of recent values may be diminished by adjusting the filter weights accordingly. In this way, rapid fluctuations of parameter values can be "smoothed" or averaged out, as appropriate, and alternatively, periodic features of the parameter values can be reduced or emphasized.

The parameter sampling intervals of the present invention may be in the form of asynchronous time intervals for measurements. Basically, fixed intervals at high rates (e.g., frequent sampling intervals) can have the effect of overtaxing the computer or processing system. Accordingly, the processor may be interrupted at normal interrupt intervals for frequency and voltage determinations. These determinations are basically performed at each activity driven interrupt (e.g., key press, touchscreen touch, etc.). The result is an asynchronous sampling interval that, in certain cases, reduces computational requirements.

The manner in which the power management module employs the various types of intervals (e.g., fixed and variable) is illustrated in FIG. 5. Specifically, the power management module determines if fixed intervals are to be employed at step 70. This may be accomplished based on various factors, such as user or OEM settings, the parameter to measure, the processor activity, etc. If a fixed interval is to be employed, the power management module selects the appropriate interval at step 72, and waits for the occurrence of that interval to perform clock frequency and voltage adjustment at step 78 as described above (e.g., FIG. 2). Otherwise, the power management module employs a variable interval (e.g., based on interrupts or other system activity) and waits for the occurrence of that interval at step 76 to perform clock frequency and voltage adjustment at step 78 as described above (e.g., FIG. 2).

The power management module may further utilize various parameters to adjust clock frequency and voltage. The parameters may be used individually as described above (e.g., for IPC), or may be combined in plural stages to determine clock frequency and voltage settings. The manner in which the power management module combines param-

8

eters in plural stages to determine clock frequency and voltage settings is illustrated in FIG. 6. Initially, the power management module is described with reference to combining thread performance zone and IPC parameters. However, the power management module may combine any desired parameters in the manner described below. The thread performance zone parameter represents a request for a particular performance level by a software thread or task executing on processor 10. This request may be provided by software (e.g., through an application programming interface or function call within the task) or hardware (e.g., via a register that stores a circuit-determined performance value based on task information). The request is presented in the form of a numerical value that represents a range from low performance to high performance. For example, the range from -1.0 to +1.0 includes -1.0 indicating lowest performance (e.g., longest battery life), 0.0 indicating average performance, and +1.0 indicating highest performance (e.g., shortest battery life). The selected default value is typically in the middle of that range (e.g., zero).

In particular, the power management module waits for a parameter sampling interval (e.g., fixed or variable) at step 80. Once the interval occurs, the performance zone parameter is retrieved at step 82 (e.g., the retrieval of the performance zone parameter represents an initial stage). Subsequently, the IPC parameter is determined at step 84 in the manner described above and added to the performance zone parameter at step 86 (e.g., this represents a subsequent stage). The net effect of the addition is to skew the IPC value towards low or high performance based on the requested performance zone. The resulting combined parameter is subsequently compared to a threshold. The threshold value is typically set to be slightly less than 1.0, typically in the range between 0.95-0.99 (e.g., threshold=1.0-Δ, where Δ is predetermined or set by a user). As discussed above, when IPC is near 1.0, the processor is operating efficiently and the highest operating frequency is utilized. Generally, the IPC parameter is typically less than 1.0 since the processor usually performs an operation in a given interval that endures for several cycles. Accordingly, the threshold accounts for these instances, and basically adjusts sensitivity of the power management module with respect to entering a high performance mode. Thus, the selection of the delta value sets the power management module sensitivity and essentially enables a trade-off between performance and battery life (e.g., the lower the delta value, the longer the battery life and the lower the performance).

When the resulting combined parameter is greater than or equal to the threshold as determined at step 88, the IPC and performance zone parameters are indicating a high performance mode and the highest frequency and corresponding voltage setting stored in memories 8 are selected at step 90 and applied at step 100 to adjust the frequency and voltage. If the combined parameter value is less than zero as determined at step 92, the IPC and performance zone parameters are indicating a power conservation mode and the lowest frequency and corresponding voltage setting stored in memories 8 are selected at step 94 and applied at step 100 to adjust the frequency and voltage. When the combined parameter is greater than zero and less than the threshold, a new frequency is determined at step 96 by multiplying the combined parameter by the current frequency. The next highest frequency within the stored frequency values is selected and the corresponding frequency and voltage settings are retrieved at step 98 and applied at step 100 to adjust the frequency and voltage. Thus, when the performance zone is 0.0, no effect is added and the IPC value and the

US 7,111,179 B1

9

current frequency value determines the new frequency value as described above. However, if the performance zone is less than or greater than 0.0, the effective IPC value is decreased or increased, causing the new frequency to skew towards lower or higher frequencies, respectively. By way of example, a processor has possible frequency settings of 100, 200, 300, and 400 MHz. With an IPC of 0.6, current frequency of 400 MHz and a performance zone of 0.0, the appropriate new frequency is 300 MHz (e.g., the next highest possible frequency value above 0.6 multiplied by 400). However, if the performance zone is -0.2, the appropriate new frequency is 200 MHz (the next highest frequency value above 0.4 multiplied by 400), while a performance zone of +0.3 maintains the frequency at 400 MHz.

The effect of or the amount of skewing by the parameters on the frequency and voltage adjustment in the plural stages may be controlled. For example, if the effect of the first stage parameter (e.g., performance zone) is to be lower relative to the second stage parameter (e.g., IPC), the range for the first parameter may be lowered (e.g., -0.5 to +0.5, instead of -1.0 to +1.0) with respect to the second parameter range. In addition, plural parameters each with a corresponding value range may be used in the first stage to skew the second stage parameter towards higher or lower frequencies. By way of example, the first stage may include two parameters, one with a range of -0.5 to +0.5 and the other with a range of -0.2 to +0.2, where the values of each of the first stage parameters are added or otherwise combined with the second stage parameter for power management determinations as described above. The plural parameters employed by the power management module may include: location, orientation, and motion of the device; interrupt-service-routines-per-cycle (ISRPC); weather/environmental conditions; wireless signal strength; and time-of-day, week, month, or year.

The plural parameters may be used individually or in any combination in stage one to skew the IPC calculation or other parameter in the subsequent stage. The device location, orientation, and motion information may be retrieved from GPS receiver 22 (FIG. 1) receiving information from a GPS system, or over a wired or wireless communications link to a network (e.g., Internet, etc.) or other source via communications device 24. These attributes are formulated as a normalized range, typically centered around 0.0. In the case of location, the parameter reflects distance from a particular location. For example, the most negative value may indicate long distances from the location, while values may increase as the device approaches that location. This has the effect of increasing performance of the device as it approaches the location. Alternatively, performance may be decreased as the location is approached, or modified in accordance with device orientation or acceleration changes.

The ISRPC value is determined by counting the number of interrupts (e.g., typically provided by processor 10) in an interval (e.g., parameter sampling interval) and dividing the count by the time interval, and further dividing the result by the clock frequency (e.g., (count/time interval)/clock frequency). However, any desired unit time interval (e.g., a desired interval, such as one millisecond, etc.) other than the clock frequency may be employed. The resulting number provides an indication of performance requirements. A higher number indicates more interrupts and generally higher processing requirements. The ISRPC value is expressed as a normalized range, typically centered around 0.0. For example, numbers greater than zero may indicate a great number of interrupts, while numbers near or below zero may indicate a small amount of interrupts.

10

The weather and environmental conditions and the ambient conditions in the environment around the device may be received from on-board sensors 26 (FIG. 1) (e.g., temperature sensors, humidity sensors, etc.) or from a wired or wireless communications link to a network (e.g., Internet, etc.) or other source via communications device 24. These attributes are formulated as a normalized range, typically centered around 0.0. For example, in the case of humidity, humidity measurements relative to a default measurement are utilized to indicate normalized values. High humidity may indicate the most negative value, while low humidity may indicate the most positive value. This has the effect of increasing performance as humidity drops. Alternatively, adjustment of performance may be based on ambient temperature or rainfall.

The relative strength of a received wireless signal (e.g., for use with wireless modems, telephones, etc.) may be used to adjust performance. The signal may be from a local wireless link, such as a cellular telephone system, an 802.11 wireless local area network or a GPS signal. In this case, the attributes are formulated as a normalized range, typically centered around 0.0. For example, as the received radio frequency signal strength increases, this may indicate that the device is approaching an area of expected increased performance, and a value above 0.0 may be set. Alternatively, as the signal strength diminishes, the value may be set below 0.0.

The actual time and date may be used to determine performance. For example, in a distributed smart sensor computer device that measures temperature every hour per day on the hour, as the measurement time approaches, the frequency and voltage may be increased by creating a normalized parameter, typically centered around 0.0, that increases above 0.0 when the time for a measurement approaches. Alternatively, the time and date information may be retrieved via an on-board real-time-clock circuit or over a wired or wireless communications link to a network (e.g., Internet, etc.) or other source via communications device 24 (FIG. 1).

The plural stage combination of parameters described above may further utilize historical measurements of any one or more of the parameters. Basically, the normalized historical measurements may be weighted and combined (e.g., as described above for FIG. 4) to produce a resulting parameter value. The resulting value may be utilized in the corresponding stage in the manner described above to determine clock frequency and voltage settings.

The power management module may further employ various graphical user interfaces (GUI) or user screens (not shown) to display and receive information from a user. For example, a user or operator may set the thresholds (or data values) or other parameters via these screens, or select desired parameters to utilize for adjusting clock frequency and voltage. Further, the screens may display any desired information, such as operating mode (e.g., frequency, voltage, general indicator indicating high performance or low power mode, etc.), enabled parameters, etc. Alternatively, this information may be displayed on an operating system screen.

It will be appreciated that the embodiments described above and illustrated in the drawings represent only a few of the many ways of implementing a method and apparatus for optimizing performance and battery life of electronic devices based on system and application parameters.

The present invention may be employed for power reduction on any type of electronic device including a processor or processing circuitry under software control (e.g., com-

US 7,111,179 B1

11

puter systems, telephones, PDAs, organizers, programmable or other calculators, etc.). These devices may utilize any quantity of any types of portable or stationary power sources (e.g., portable, wall outlet, batteries, solar, fuel cells, etc.), however the present invention is preferably utilized for extending the operating life of devices with portable power sources.

The computer system employing the present invention may be implemented by any quantity of any personal or other type of computer or processing system (e.g., IBM-compatible, Apple, Macintosh, laptop, palm pilot, microprocessor, etc.). The computer system may include any commercially available operating system (e.g., Windows, OS/2, Unix, Linux, etc.), any commercially available and/or custom software (e.g., communications software, power management module software, etc.) and any types of input devices (e.g., keyboard, mouse, microphone, voice recognition, etc.). The various computer system components (e.g., memories, power supply, processor, peripherals, buses, etc.) are typically implemented by any quantity of conventional components and/or devices, where their configurations may vary slightly to accommodate particular characteristics of a device. The processor may be implemented by any conventional or other microprocessor or processing device (e.g., such as those available from Intel). It is to be understood that the software of the present invention (e.g., power management module, etc.) may be implemented in any desired computer language, and could be developed by one of ordinary skill in the computer and/or programming arts based on the functional description contained herein and the flow charts illustrated in the drawings. Further, any references herein of software performing various functions generally refer to computer systems or processors performing those functions under software control.

The power management module may alternatively be implemented by any quantity of hardware or other processing circuitry (e.g., ASIC, PGA, IC, logic or other circuitry, etc.) and may be integrated within a processor to form a new processor or be external of the processor (e.g., on or external to the processor board) and coupled to the frequency and voltage adjustment modules. The various functions of the power management module may be distributed in any manner among any quantity (e.g., one or more) of hardware and/or software modules or units, computer or processing systems or circuitry, where the computer or processing systems may be disposed locally or remotely of each other and communicate via any suitable communications medium (e.g., LAN, WAN, Intranet, Internet, hardwire, modem connection, wireless, etc.). The software and/or algorithms described above and illustrated in the flow charts and drawings may be modified in any manner that accomplishes the functions described herein.

The power management module may be implemented as a separate stand-alone software or hardware module or may be in the form of an embeddable system (e.g., within another system, as an embeddable software component within other software, etc.). The power management module software may be available on a recorded medium (e.g., magnetic or optical mediums, magneto-optic mediums, floppy diskettes, CD-ROM, memory devices, etc.) for use on stand-alone systems or systems connected by a network or other communications medium, and/or may be downloaded (e.g., in the form of carrier waves, packets, etc.) to systems via a network or other communications medium.

The processor may include any quantity of any types of peripherals to ascertain parameter information. The GPS receiver may be implemented by any quantity of conven-

12

tional or other receivers capable of communicating with a GPS system. The communications device may be implemented by any quantity of any conventional communications devices (e.g., wireless or wired modem, wired or wireless network card, etc.) to interface a network (e.g., Internet, etc.) or other source of information. The sensors may be of any quantity and include any type of conventional or other sensors to measure any quantity of desired parameters (e.g., temperature sensor, humidity sensor, rain sensor, etc.). The peripherals may basically ascertain any desired information for the power management determinations either by accessing an information source or by direct measurement.

The reduction of power may be accomplished by any desired techniques (e.g., adjusting frequency and voltage, disabling power to components, etc.). The control signals to adjust frequency and voltage may be of any quantity, may be in any desired format and may contain any desired information. The power management module may communicate directly with the frequency adjustment and voltage adjustment modules or indirectly via the processor. The frequency and voltage may be adjusted individually or in combination based on parameter determinations, where the parameters may relate solely to voltage or frequency, or to both. The frequency and voltage may be adjusted in any desired order and during the same or different intervals.

The type of intervals may be selected in any desired fashion and based on any conditions (e.g., parameters to be measured, system activity, processing efficiency, user preferences or settings, etc.). The fixed intervals may be set to any desired time intervals (e.g., on the order of milliseconds, microseconds, seconds, etc.). The fixed intervals may be predetermined or set by a user or operator. The variable intervals may be based on any desired system activities (e.g., interrupts, thread or task switching, peripherals, activities indicated by a user or operator, etc.). The power management module may include and/or monitor any quantity of conventional or other devices to monitor the fixed and/or variable intervals (e.g., counters, timers, polling registers, etc.).

The present invention may store and/or utilize parameter samples from any quantity of sampling intervals (e.g., current and previous). Further, any quantity of samples for a parameter may be obtained in a sampling interval, and individual parameters may be measured during each interval or during particular intervals (e.g., every Nth interval, where N is an integer, during intervals triggered by a specific activity, etc.). The parameter values may be stored in memory in any desired fashion (e.g., an array, list, table, etc.). The frequency and voltage settings may be stored in the memories in any desired fashion (e.g., an array, list, table, etc.) and may include any quantity of settings in any desired increments or units. Moreover, the memories may store settings and/or controls for any desired device components to reduce power consumption for those components. The memories may include any quantity of any types of memory (e.g., ROM, RAM, etc.), while the various parameters and settings may be stored in any device internal or external memories.

The IPC may be determined in any desired manner based on any desired unit time interval (e.g., clock cycle, any time interval, such as milliseconds, microseconds, etc.). The instruction and cycle count may be determined by the processor, or may be determined by the power management module (e.g., counting instructions and cycles, calculating the number of cycles based on the clock frequency and sampling interval, etc.). The IPC and other parameters may

US 7,111,179 B1

13

utilize any desired numeric or other range (e.g., symbols or characters with numeric assignments, etc.). The parameter value may be combined with the current frequency or any reference frequency in any desired manner (e.g., multiplied, added, indicate a multiplier, etc.) to attain a new frequency setting. Further, a new frequency setting may be selected from the stored settings in any desired fashion (e.g., select the next highest or lowest, select a frequency based on an offset or a positional offset (e.g., select the second highest or lowest frequency) from the calculated or a reference frequency, etc.).

Any quantity of current and/or prior measurements for a parameter may be combined (e.g., weighted average, filtered, average, accumulated, etc.) in any fashion to produce an overall result. The weights may be of any desired values to achieve a desired effect of values on the overall parameter value. Any quantity of any types of parameters may be combined in any quantity of stages for power management determinations, where each stage may utilize any quantity or type of parameter. The parameter values may be normalized with respect to any desired value range (e.g., numeric, characters or symbols assigned numeric values, etc.). The parameter ranges may be set to any desired values to control the influence of a parameter in the power management determination. The parameter values may be combined in any fashion (e.g., added, multiplied, etc.). Alternatively, the parameter values may be utilized to select or modify thresholds for entering high performance or power conservation modes. For example, the plural parameters may be utilized to adjust the threshold for IPC values to control entry into high performance or power conservation modes.

The thread performance zone parameter may be indicated by software and/or hardware and may include any desired range to indicate performance. This parameter may request any desired specific operating characteristic of the device (e.g., specific frequency or voltage, etc.). The thread or task may further request any desired characteristic in any desired fashion (e.g., request a device to be disabled, etc.).

The ISRPC may be determined in any desired manner based on any desired unit time interval (e.g., clock cycle, any time interval, such as milliseconds, microseconds, etc.). The interrupt cycle count may be determined by the processor, or may be determined by the power management module (e.g., counting interrupts, etc.). The various plural parameters (e.g., location, motion, weather, etc.) may be obtained from external information sources (e.g., Internet, GPS system, etc.) or direct sensor measurement in any desired fashion.

The GUI or user screens may be of any quantity, may be arranged in any desired fashion and may include any desired information (e.g., indicate mode, enabled parameters, etc.). The screens may receive any information from a user (e.g., frequency, voltage or other system device settings to control the device, parameters to utilize for power management determinations, time for fixed intervals, activity for variable intervals, etc.). The information may alternatively be displayed and/or received from any quantity of screens of other software (e.g., operating system, application program, etc.) or any combination of the user and other software screens.

The techniques described above for examining parameters for power management decisions may be used by the present invention individually, or in any combination to control power usage of the device. Further, each of these techniques may employ fixed or variable intervals or any combinations thereof, and may utilize any desired parameter or combinations of parameters to render power management determinations.

14

It is to be understood that the present invention is not limited to the applications disclosed herein, but may be utilized with any electronic devices including a processor to reduce device power consumption. Further, the present invention may measure and analyze any desired device or external parameters to reduce device power consumption or control other device operating characteristics.

From the foregoing description, it will be appreciated that the invention makes available a novel method and apparatus for optimizing performance and battery life of electronic devices based on system and application parameters, wherein an electronic device monitors system and application parameters to control device power usage.

Having described preferred embodiments of a new and improved method and apparatus for optimizing performance and battery life of electronic devices based on system and application parameters, it is believed that other modifications, variations and changes will be suggested to those skilled in the art in view of the teachings set forth herein. It is therefore to be understood that all such variations, modifications and changes are believed to fall within the scope of the present invention as defined by the appended claims.

What is claimed is:

1. A system for dynamically managing power within an electronic device comprising:

a power management device including:

a sampling module to ascertain and store a plurality of parameter values associated with said electronic device during at least one sampling interval;

an analysis module to combine said plurality of parameter values to produce an overall parameter value and to evaluate said overall parameter value to determine at least one control value to adjust power consumption of said electronic device in response to said parameter evaluation, wherein said plurality of parameters are arranged into a series of successive stages with each stage including at least one parameter, and said analysis module includes:

a normalization module to normalize said parameter values relative to a numeric range; and

a combination module to mathematically combine said normalized parameters of each subordinate stage with a succeeding stage to produce an overall value to determine said at least one control value to adjust said power consumption of said electronic device; and

an adjustment module to adjust power consumption of said electronic device in accordance with said at least one control value to control electronic device power consumption.

2. The system of claim 1, wherein said sampling module ascertains and stores values for a parameter for a plurality of sampling intervals, and said analysis module includes:

a history module to determine said at least one control value based on a parameter value from a current sampling interval and at least one parameter value from a prior sampling interval to adjust said power consumption of said electronic device.

3. The system of claim 1, wherein said sampling module selects a type of sampling interval from among a series of sampling interval types in accordance with electronic device conditions, wherein said series of sampling interval types include fixed and variable sampling intervals.

4. The system of claim 3, wherein said selected sampling interval is a fixed sampling interval that occurs periodically at a prescribed time interval.

US 7,111,179 B1

15

5. The system of claim 3, wherein said sampling interval is a variable sampling interval that commences in response to occurrence of a particular electronic device activity.

6. The system of claim 5, wherein said electronic device activity includes an activity driven interrupt of an electronic device processor.

7. The system of claim 1, wherein said at least one control value includes at least one of an operating frequency and a power level to utilize for an electronic device component.

8. A system for dynamically managing power within an electronic device comprising:

a power management device including:

a sampling module to ascertain and store a plurality of parameter values associated with said electronic device during at least one sampling interval, wherein said plurality of parameters relates to at least two of: a quantity of instructions executed per time interval for an electronic device processor; an interrupt density for said electronic device processor; a performance request by a thread executing on said electronic device processor; weather and/or environmental conditions; location, orientation and/or motion of said electronic device; a signal strength of a wireless signal received by said electronic device; a time of day; and date information;

an analysis module to determine at least one control value based on said plurality of parameter values to adjust power consumption of said electronic device; and

an adjustment module to adjust power consumption of said electronic device in accordance with said at least one control value to control electronic device power consumption.

9. The system of claim 8, wherein said sampling module selects a type of sampling interval from among a series of sampling interval types in accordance with electronic device conditions, wherein said series of sampling interval types include fixed and variable sampling intervals.

10. The system of claim 9, wherein said selected sampling interval is a fixed sampling interval that occurs periodically at a prescribed time interval.

11. The system of claim 9, wherein said sampling interval is a variable sampling interval that commences in response to occurrence of a particular electronic device activity.

12. The system of claim 11, wherein said electronic device activity includes an activity driven interrupt of an electronic device processor.

13. The system of claim 8, wherein said analysis module includes:

a history module to determine said at least one control value based on a parameter value from a current sampling interval and at least one parameter value from a prior sampling interval to adjust said power consumption of said electronic device.

14. The system of claim 8, wherein said at least one control value includes at least one of an operating frequency and a power level to utilize for an electronic device component.

15. A system for dynamically managing power within an electronic device comprising:

a power management device including:

a sampling module to ascertain and store a plurality of parameter values associated with said electronic device during at least one sampling interval, wherein said sampling module ascertains and stores values for a parameter for a plurality of sampling intervals;

16

an analysis module to combine said plurality of parameter values to produce an overall parameter value and to evaluate said overall parameter value to determine at least one control value to adjust power consumption of said electronic device in response to said parameter evaluation, wherein said analysis module includes a combination module to combine a parameter value from a current sampling interval with at least one parameter value from a prior sampling interval to produce an overall value to determine said at least one control value to adjust said power consumption of said electronic device, and wherein said combination module includes a parameter module to determine a weighted average of said current parameter value and said at least one prior parameter value to produce said overall value to determine said at least one control value to adjust said power consumption of said electronic device; and

an adjustment module to adjust power consumption of said electronic device in accordance with said at least one control value to control electronic device power consumption.

16. The system of claim 15, wherein said sampling module selects a type of sampling interval from among a series of sampling interval types in accordance with electronic device conditions, wherein said series of sampling interval types include fixed and variable sampling intervals.

17. The system of claim 16, wherein said selected sampling interval is a fixed sampling interval that occurs periodically at a prescribed time interval.

18. The system of claim 16, wherein said sampling interval is a variable sampling interval that commences in response to occurrence of a particular electronic device activity.

19. The system of claim 18, wherein said electronic device activity includes an activity driven interrupt of an electronic device processor.

20. The system of claim 15, wherein said at least one control value includes at least one of an operating frequency and a power level to utilize for an electronic device component.

21. A method of dynamically managing power within an electronic device comprising the steps of:

(a) ascertaining and storing a plurality of parameter values associated with said electronic device during at least one sampling interval;

(b) combining said plurality of parameter values to produce an overall parameter value and evaluating said overall parameter value to determine at least one control value to adjust power consumption of said electronic device in response to said parameter evaluation, wherein said plurality of parameters are arranged into a series of successive stages with each stage including at least one parameter, and step (b) further includes: normalizing said parameter values relative to a numeric range; and

mathematically combining said normalized parameters of each subordinate stage with a succeeding stage to produce an overall value to determine said at least one control value to adjust said power consumption of said electronic device; and

(c) adjusting power consumption of said electronic device in accordance with said at least one control value to control electronic device power consumption.

22. The method of claim 21, wherein step (a) further includes:

US 7,111,179 B1

17

ascertaining and storing values for a parameter for a plurality of sampling intervals; and
step (b) further includes:

determining said at least one control value based on a parameter value from a current sampling interval and at least one parameter value from a prior sampling interval to adjust said power consumption of said electronic device.

23. The method of claim 21, wherein step (a) further includes:

selecting a type of sampling interval from among a series of sampling interval types in accordance with electronic device conditions, wherein said series of sampling interval types include fixed and variable sampling intervals.

24. The method of claim 23, wherein step (a) further includes:

selecting a fixed sampling interval that occurs periodically at a prescribed time interval.

25. The method of claim 23, wherein step (a) further includes:

selecting a variable sampling interval that commences in response to occurrence of a particular electronic device activity.

26. The method of claim 25, wherein said electronic device activity includes an activity driven interrupt of an electronic device processor.

27. The method of claim 21, wherein step (b) further includes:

determining at least one of an operating frequency and a power level to utilize for an electronic device component to adjust power consumed by said electronic device.

28. A method of dynamically managing power within an electronic device comprising the steps of:

(a) ascertaining and storing a plurality of parameter values associated with said electronic device during at least one sampling interval, wherein said plurality of parameters relates to at least two of: a quantity of instructions executed per time interval for an electronic device processor; an interrupt density for said electronic device processor; a performance request by a thread executing on said electronic device processor; weather and/or environmental conditions; location, orientation and/or motion of said electronic device; a signal strength of a wireless signal received by said electronic device; a time of day; and date information;

(b) determining at least one control value based on said plurality of parameter values to adjust power consumption of said electronic device; and

(c) adjusting power consumption of said electronic device in accordance with said at least one control value to control electronic device power consumption.

29. The method of claim 28, wherein step (a) further includes:

selecting a type of sampling interval from among a series of sampling interval types in accordance with electronic device conditions, wherein said series of sampling interval types include fixed and variable sampling intervals.

30. The method of claim 29, wherein step (a) further includes:

selecting a fixed sampling interval that occurs periodically at a prescribed time interval.

31. The method of claim 29, wherein step (a) further includes:

18

selecting a variable sampling interval that commences in response to occurrence of a particular electronic device activity.

32. The method of claim 31, wherein said electronic device activity includes an activity driven interrupt of an electronic device processor.

33. The method of claim 28, wherein step (a) further includes:

ascertaining and storing values for a parameter for a plurality of sampling intervals; and

step (b) further includes:

determining said at least one control value based on a parameter value from a current sampling interval and at least one parameter value from a prior sampling interval to adjust said power consumption of said electronic device.

34. The method of claim 28, wherein step (b) further includes:

determining at least one of an operating frequency and a power level to utilize for an electronic device component to adjust power consumed by said electronic device.

35. A method of dynamically managing power within an electronic device comprising the steps of:

(a) ascertaining and storing a plurality of parameter values associated with said electronic device during at least one sampling interval, wherein step (a) further includes:

ascertaining and storing values for a parameter for a plurality of sampling intervals;

(b) combining said plurality of parameter values to produce an overall parameter value and evaluating said overall parameter value to determine at least one control value to adjust power consumption of said electronic device in response to said parameter evaluation, wherein step (b) further includes:

determining a weighted average of a parameter value from a current sampling interval and at least one parameter value from a prior sampling interval to produce said overall value to determine said at least one control value to adjust said power consumption of said electronic device; and

(c) adjusting power consumption of said electronic device in accordance with said at least one control value to control electronic device power consumption.

36. The method of claim 35, wherein step (a) further includes:

selecting a type of sampling interval from among a series of sampling interval types in accordance with electronic device conditions, wherein said series of sampling interval types include fixed and variable sampling intervals.

37. The method of claim 36, wherein step (a) further includes:

selecting a fixed sampling interval that occurs periodically at a prescribed time interval.

38. The method of claim 36, wherein step (a) further includes:

selecting a variable sampling interval that commences in response to occurrence of a particular electronic device activity.

39. The method of claim 38, wherein said electronic device activity includes an activity driven interrupt of an electronic device processor.

40. The method of claim 35, wherein step (b) further includes:

US 7,111,179 B1

19

determining at least one of an operating frequency and a power level to utilize for an electronic device component to adjust power consumed by said electronic device.

41. A system for dynamically managing power within an electronic device comprising:

a power management device including:

a sampling module to ascertain a value for at least one parameter associated with said electronic device during a sampling interval, wherein said at least one parameter includes a quantity of instructions executed by an electronic device processor within a particular time interval;

an analysis module to evaluate said at least one parameter value and determine at least one control value to adjust power consumption of said electronic device in response to said parameter evaluation, wherein said at least one control value includes an operating frequency and voltage of said electronic device processor, and wherein said analysis module includes: an evaluation module to set at least one control value to a maximum operating frequency of said electronic device processor in response to said instruction quantity parameter value being equal to or exceeding a threshold and to set at least one control value to a desired operating frequency for said electronic device processor derived from the product of the current processor operating frequency and said instruction quantity parameter value in response to said instruction quantity parameter value being less than said threshold;

an adjustment module to adjust power consumption of said electronic device in accordance with said at least one control value to control electronic device power consumption.

20

42. A method of dynamically managing power within an electronic device comprising the steps of:

(a) ascertaining a value for at least one parameter associated with said electronic device during a sampling interval, wherein said at least one parameter includes a quantity of instructions executed by an electronic device processor within a particular time interval;

(b) evaluating said at least one parameter value and determining at least one control value to adjust power consumption of said electronic device in response to said parameter evaluation, wherein step (b) further includes:

determining an operating frequency and voltage of an electronic device processor to adjust power consumption of said electronic device in response to said parameter evaluation;

setting at least one control value to a maximum operating frequency of said electronic device processor in response to said instruction quantity parameter value being equal to or exceeding a threshold; and

setting at least one control value to a desired operating frequency for said electronic device processor derived from the product of the current processor operating frequency and said instruction quantity parameter value in response to said instruction quantity parameter value being less than said threshold; and

(c) adjusting power consumption of said electronic device in accordance with said at least one control value to control electronic device power consumption.

* * * * *

EXHIBIT B



US007051306B2

(12) **United States Patent**
Hoberman et al.

(10) **Patent No.:** **US 7,051,306 B2**
(45) **Date of Patent:** **May 23, 2006**

(54) **MANAGING POWER ON INTEGRATED CIRCUITS USING POWER ISLANDS**

6,667,648 B1 * 12/2003 Stout et al. 327/333
6,779,163 B1 * 8/2004 Bednar et al. 716/7
6,820,240 B1 * 11/2004 Bednar et al. 716/1

(75) Inventors: **Barry Alan Hoberman**, Cupertino, CA (US); **Daniel L. Hillman**, San Jose, CA (US); **Jon Shiell**, San Jose, CA (US)

(73) Assignee: **MOSAID Technologies Corporation**, Sunnyvale, CA (US)

(*) Notice: Subject to any disclaimer, the term of this patent is extended or adjusted under 35 U.S.C. 154(b) by 0 days.

(21) Appl. No.: **10/840,893**

(22) Filed: **May 7, 2004**

(65) **Prior Publication Data**

US 2004/0268278 A1 Dec. 30, 2004

Related U.S. Application Data

(60) Provisional application No. 60/468,742, filed on May 7, 2003.

(51) **Int. Cl.**

G06F 17/50 (2006.01)

(52) **U.S. Cl.** **716/6; 716/1; 716/4; 716/5**

(58) **Field of Classification Search** **716/6, 716/5, 4, 1**

See application file for complete search history.

(56) **References Cited**

U.S. PATENT DOCUMENTS

6,462,976 B1 * 10/2002 Olejniczak et al. 363/147
6,598,148 B1 7/2003 Moore et al.
6,631,502 B1 * 10/2003 Buffet et al. 716/4

OTHER PUBLICATIONS

Lackey et al. "Managing Power and Performance for System-on-Chip Design using Voltage Islands", Nov. 2002, IEEE International Conference on Computer Aided Design, pp. 195-202.*

"Solving the Power Management Dilemma in System-On-Chip Asic Design", Apr. 2002, IBM Technology Group.*
Semeraro et al., "Dynamic Frequency and Voltage Control for a Multiple Clock Domain Microarchitecture", Nov. 2002, IEEE 35th Annual International Symposium on Microarchitecture, Proceedings pp. 356-367.*

Akui et al., "Dynamic Voltage and Frequency Management for Low-Power Embedded Microprocessor", Feb. 2004, IEEE International Solid-State Circuits Conference, Digest of Technical Papers, vol. 1, pp. 64-513.*

(Continued)

Primary Examiner—Sun James Lin

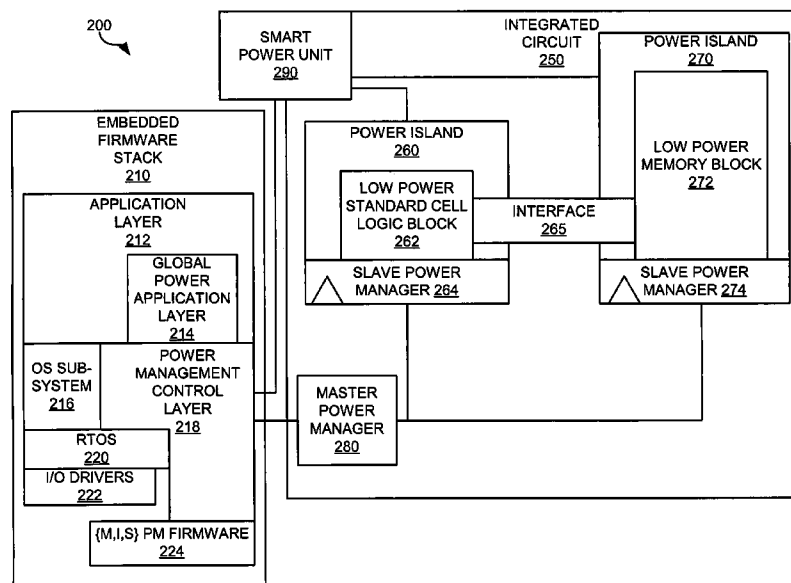
(74) *Attorney, Agent, or Firm*—Carr & Ferrell LLP

(57)

ABSTRACT

Systems and methods manage power in an integrated circuit using power islands. The integrated circuit includes a plurality of power islands where power consumption is independently controlled within each of the power islands. A power manager determines a target power level for one of the power islands. The power manager then determines an action to change a consumption power level of the one of the power islands to the target power level. The power manager performs the action to change the consumption power level of the one of the power islands to the target power level.

51 Claims, 13 Drawing Sheets



US 7,051,306 B2

Page 2

OTHER PUBLICATIONS

- Talpes et al., "Toward a Multiple Clock/Voltage Island Design Style for Power-Aware Processors", May 2005, IEEE Transactions on Very Large Scale Integration (VLSI) system, vol. 13, iss. 5, pp. 591-603.*
- U.S. Appl. No. 10/996,739, Randy J. Caplan, Systems and Methods for Minimizing Static Leakage of an Integrated Circuit, Nov. 24, 2004.
- Gupta, R., et al., "Low Power Wireless Networked System Design," HotChips, Aug. 18, 2002, Stanford, CA.
- Tsai, Y-F et al., "Implications of Technology Scaling on Leakage Reduction Techniques," DAC 2003, Jun. 2-6, 2003, ACM, Anaheim, CA, USA.
- Li L. et al., "Managing Leakage Energy in Cache Hierarchies," Journal of Instruction-Level Parallelism, 2003, pp. 1-24, vol. 5.
- Hu, J.S. et al., "Using Dynamic Branch Behavior for Power-Efficient Instruction Fetch," Proc. of IEEECS Annual Symposium on VLSI (ISVLSI 2003), Feb. 20-21, 2003, Tampa, FL.
- Zhang, W. et al., "Compiler Support for Reducing Leakage Energy Consumption," Proceedings of the 6th Design Automation and Test in Europe Conference (DATE-03), Mar., 2003, Munich, Germany.
- Degalahal, V. et al., "Analyzing Soft Errors in Leakage Optimized SRAM Design," Proceedings of the 16th International Conference on VLSI Design, Jan. 2003, New Delhi, India.
- Li, L. et al., "Leakage Energy Management in Cache Hierarchies," 11th International Conference on Parallel Architectures and Compilation Techniques (PACT'02), Sep. 2002.
- Duarte, D. et al., "Impact of Scaling on the Effectiveness of Dynamic Power Reduction Schemes," Proceedings of the 20th International Conference on Computer Design (ICCD), Sep. 16-18, 2002, Freiberg, Germany.
- Duarte, D. et al., "Impact of Technology Scaling and Packaging on Dynamic Voltage Scaling Techniques," Proceedings of the 15th Annual IEEE International ASIC/SOC Conference, Sep. 25-28, 2002, Rochester, NY.
- Kim, S et al., "Predictive Precharging for Bitline Leakage Energy Reduction," Proceedings of 15th Annual IEEE International ASIC/SOC Conference, Sep. 25-28, 2002, Rochester, NY.
- Ramanarayanan, R. et al. "Characterizing Dynamic and Leakage Power Behavior in Flip Flops," Proceedings of the 15th Annual IEEE International ASIC/SOC Conference, Sep. 2002.
- Chen, G. et al. "Adaptive Garbage Collection for Battery-Operated Environments," Proceedings of USENIX JVM02 Symposium, Aug. 2002.
- Chen, G. et al. "Energy Savings Through Compression in Embedded Java Environments," Proceedings of the CODES'02, Jun. 2002.
- Delaluz, V. et al. "Hardware and Software Techniques for Controlling DRAM Power Modes," IEEE Transaction on Computers, Nov. 2001, vol. 50.
- Delaluz, V. et al. "Scheduler-Based DRAM Energy Management," DAC 2002, Jun. 10-14, 2002, New Orleans, LA, USA, ACM.
- Kim, S. et al. "Partitioned Instruction Cache Architecture for Energy Efficiency" ACM Transactions on Computational Logic, Jul. 2002, pp 1-23, vol. V, No. N.
- De La Luz, V. et al. "Automatic Data Migration for Reducing Energy Consumption in Multi-Bank Memory Systems," DAC 2002, Jun. 10-14, 2002, New Orleans, LA, USA.
- Duarte, D. et al. "Impact of Technology Scaling in the Clock System Power," Proceedings of the IEEE Computer Society Annual Symposium on VLSI, Apr. 25-26, 2002, Pittsburgh, PA.
- Duarte, D. et al. "A Complete Phase-Locked Loop Power Consumption Model," Proceedings of the 2002 Design, Automation and Test in Europe Conference and Exhibition (DATE'02), 2002, IEEE.
- Duarte, D. et al. "Evaluating Run-Time Techniques for Leakage Power Reduction," Proceedings of the 15th International Conference on VLSI Design (VLSID'02) 2002, IEEE.
- Zhang, W. "Exploiting VLIW Schedule Slacks for Dynamic and Leakage Energy Reduction," Proceedings of the 34th Annual International Symposium on Microarchitecture (MICRO'01), Dec. 2001.
- Kim, S. et al. "Energy Efficient Instruction Cache Using Page-Based Placement," Proceedings of International Conference on Compilers, Architectures and Synthesis for Embedded Systems, CASES'01, Nov. 1-17, 2001, Atlanta, GA, USA.
- Hezavei, J. et al. "Input Sensitive High-Level Power Analysis," Proceedings of the 2001 IEEE Workshop on Signal Processing Systems (SiPS), Sep. 2001, pp. 149-156.
- Kim, S. et al. "Power-aware Partitioned Cache Architectures," Proceedings of ACM/IEEE International Symposium on Low Power Electronics and Design, 2001, ILSPED'01, Aug. 6-7, 2001, Huntington Beach, CA.
- Delaluz, V. "DRAM Energy Management Using Software and Hardware Directed Power Mode Control," Proceedings of the 7th International Symposium on High Performance Computer Architecture, Jan. 20-24, 2001, Monterrey, Mexico.
- De Micheli, G. et al. "System-Level Power Optimization: Techniques and Tools," DATE 2000.
- Manne, S. et al. "Cool Chips Tutorial," 32nd Annual International Symposium on Microarchitecture, Nov. 15, 1999, Haifa, Israel.
- Filseth, E. "Tally Power into Cost of 'Free' Silicon," EETimes, Jan. 11, 1999, located at <http://www.eetimes.com/story/speakout/OEG19990111S028>.
- Frenkil, J. "A Multi-Level Approach to Low-Power IC Design," IEEE Spectrum, Feb. 1998, Vol. 35, No. 2.
- "Managing Power in Ultra Deep Submicron ASIC/IC Design," May 2002, Synopsys, Inc.
- Pangrle, B. "Low Power Design: A Holistic Approach in an Era of New Semiconductor Technologies," San Diego Telecom Council Semiconductor SIG, Nov. 14, 2002.
- Rabaey, J. "Design Aids for Low Power: Part II Architecture and System Levels," 1997.
- Grunwald, D. et al. "Kool Chips Workshop," MICRO33, Dec. 10, 2000, Monterey, CA.
- Flynn, J. et al., "Power Management in Complex SoC Design," Synopsys, Apr. 2004, located at <http://www.synopsys.com/sps>.
- Halter et al., "A Gate-Level Leakage Power Reduction Method for Ultra-Low-Power CMOS Circuits", in Proc. IEEE Custom Integrated Circuits Conf., 1997, pp. 475-478.
- Horiguchi et al., "Switched-Source-Impedance CMOS Circuit for Low Standby Subthreshold Current Giga-Scale LSI's", IEEE Journal of Solid-State Circuits, Nov. 1993, vol. 28, No. 11.
- Kao et al., "Dual-Threshold Voltage Techniques for Low-Power Digital Circuits", IEEE Journal of Solid-State Circuits, Jul. 2000, vol. 35, No. 7.
- Mutoh, "1-V Power Supply High-Speed Digital Circuit Technology with Multithreshold-Voltage CMOS", IEEE Journal of Solid-State Circuits, Aug. 1995, vol. 30, No. 8.

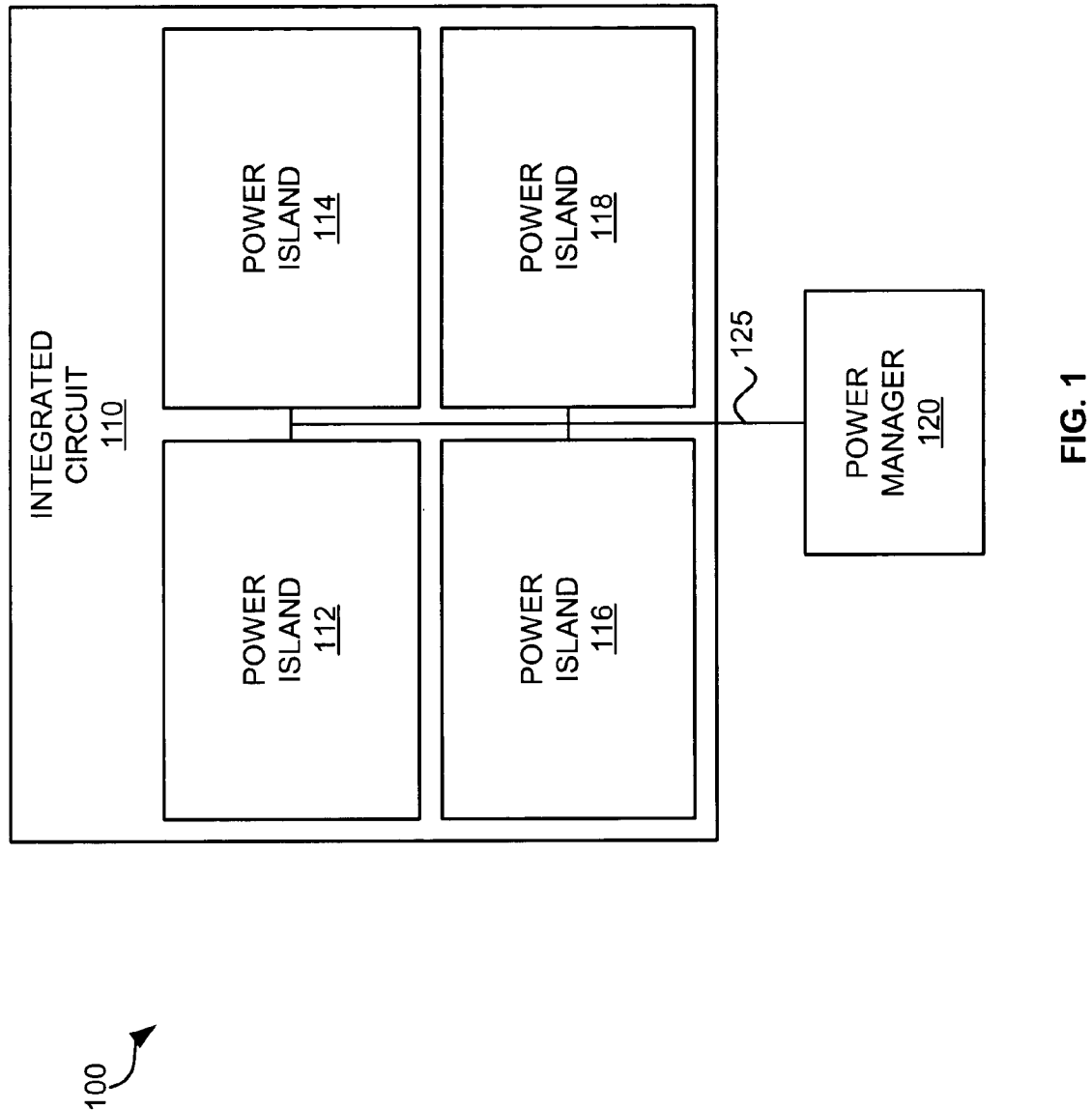
* cited by examiner

U.S. Patent

May 23, 2006

Sheet 1 of 13

US 7,051,306 B2



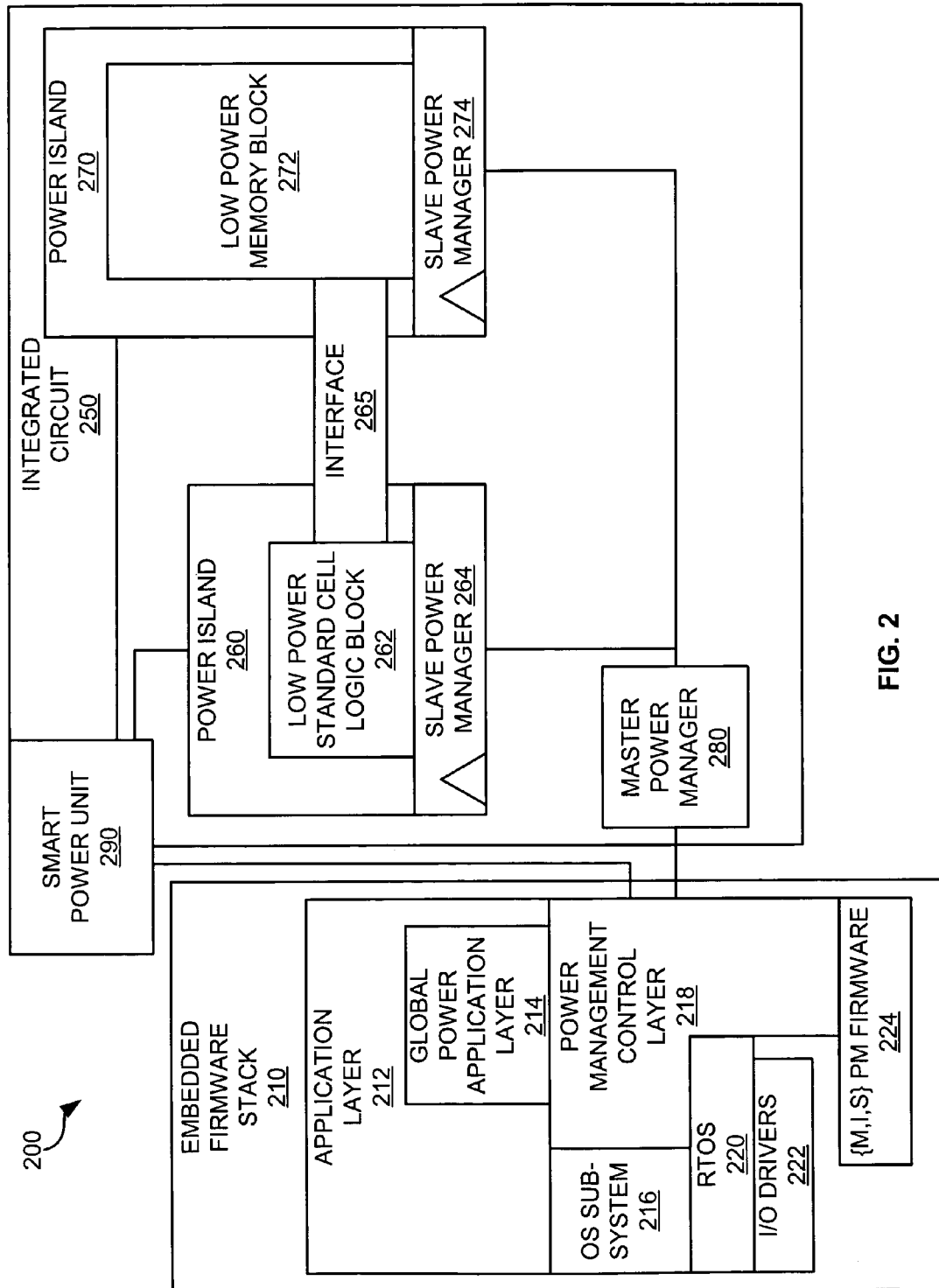


FIG. 2

U.S. Patent

May 23, 2006

Sheet 3 of 13

US 7,051,306 B2

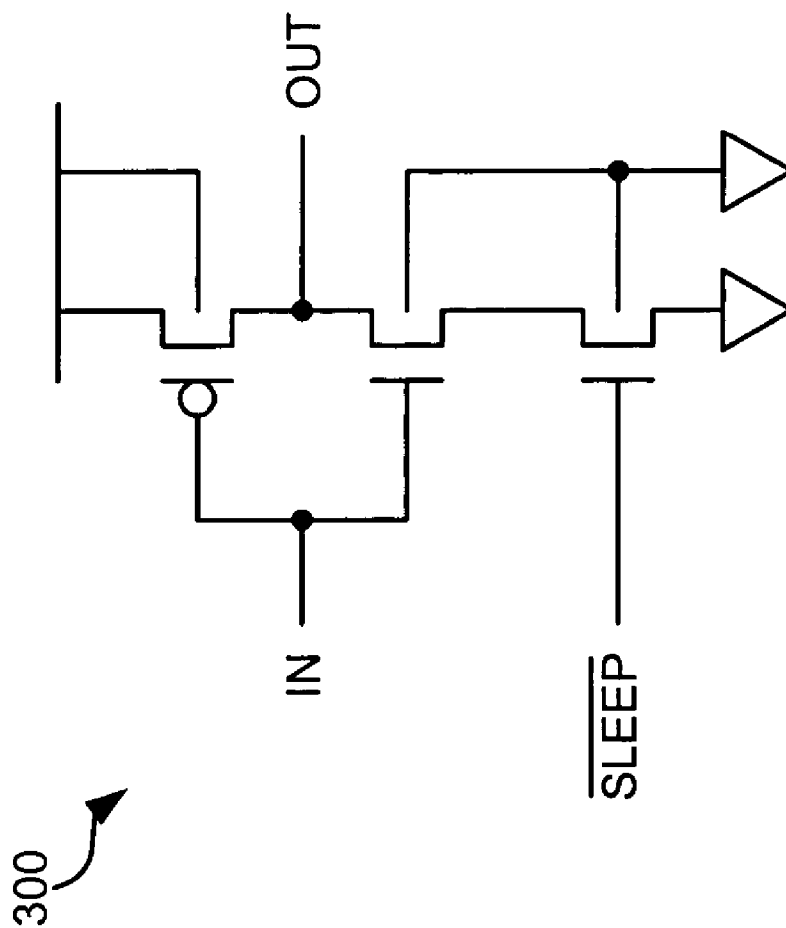


FIG. 3

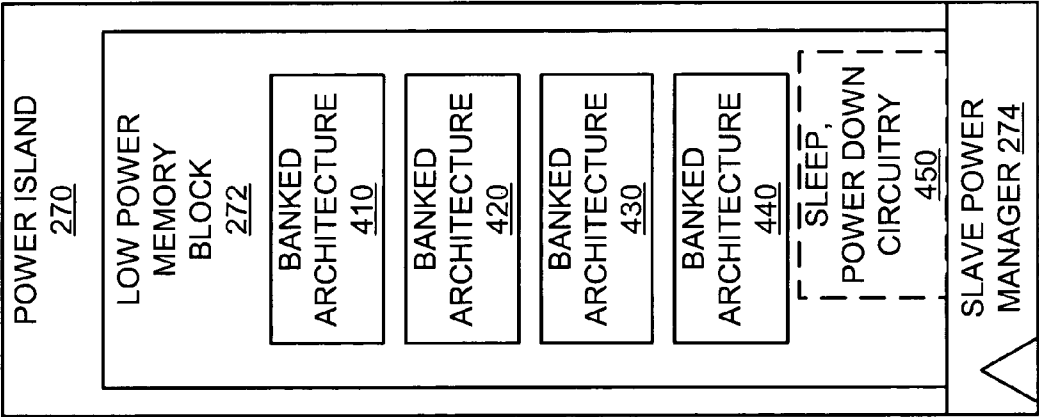


FIG. 4

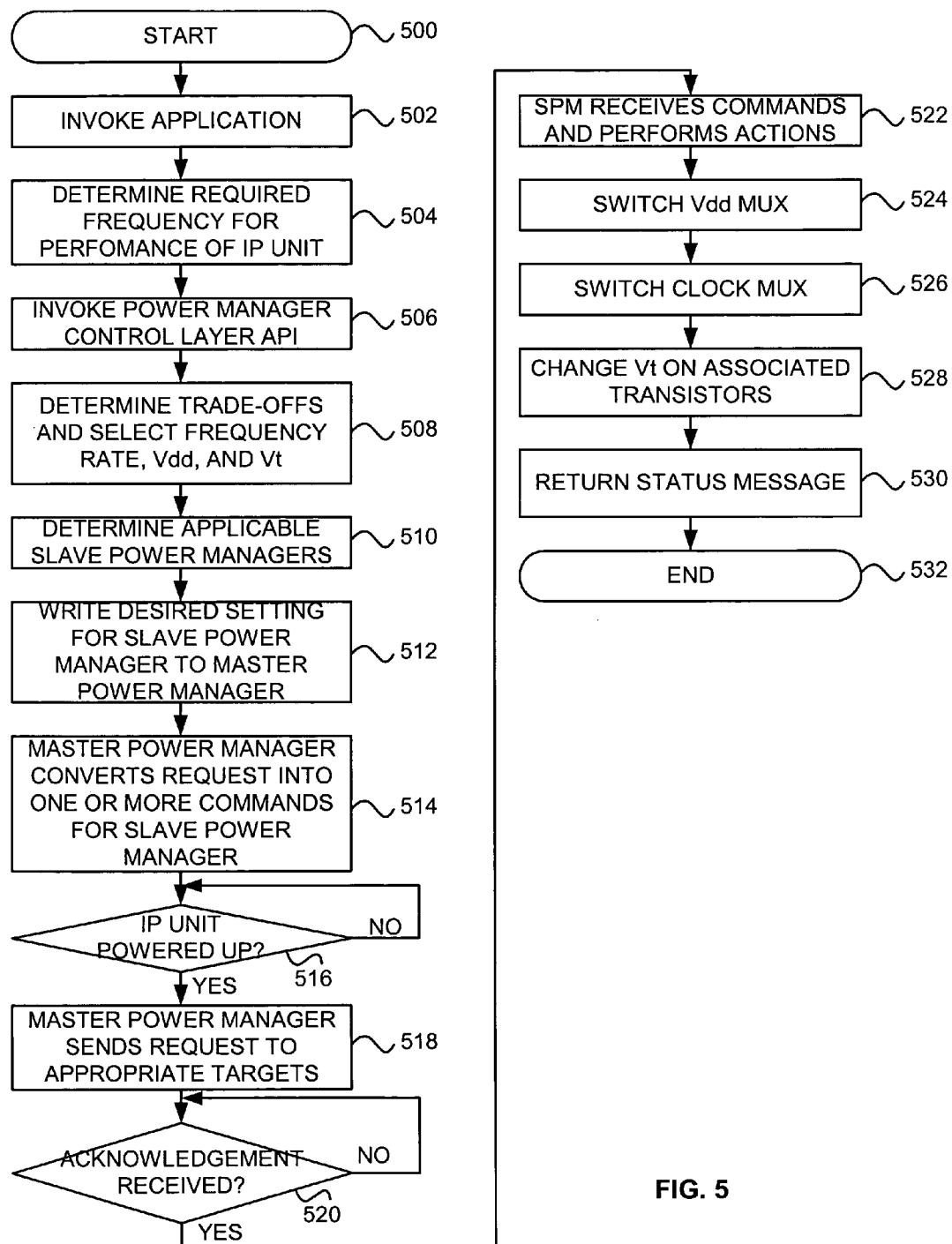


FIG. 5

U.S. Patent

May 23, 2006

Sheet 6 of 13

US 7,051,306 B2

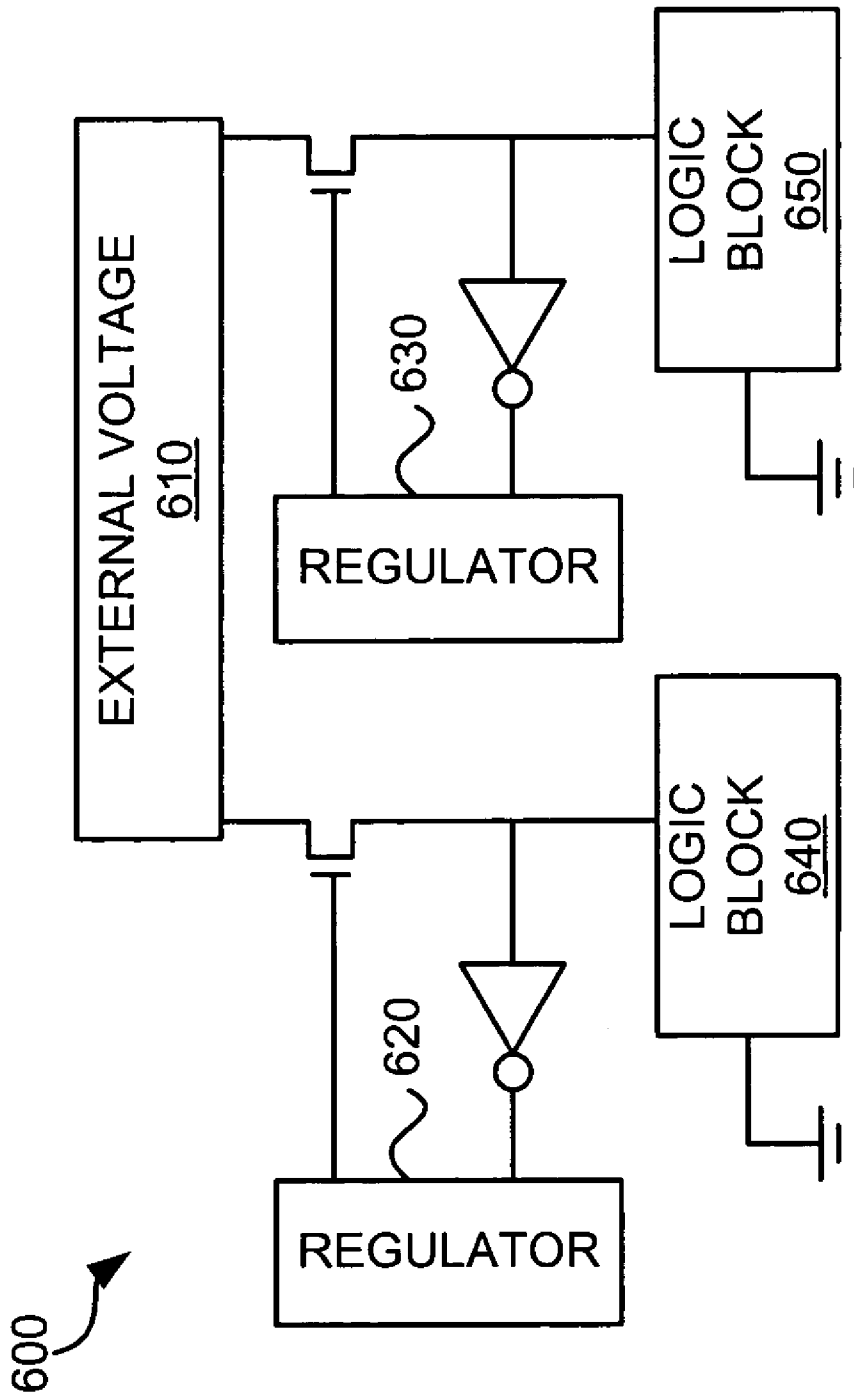


FIG. 6

U.S. Patent

May 23, 2006

Sheet 7 of 13

US 7,051,306 B2

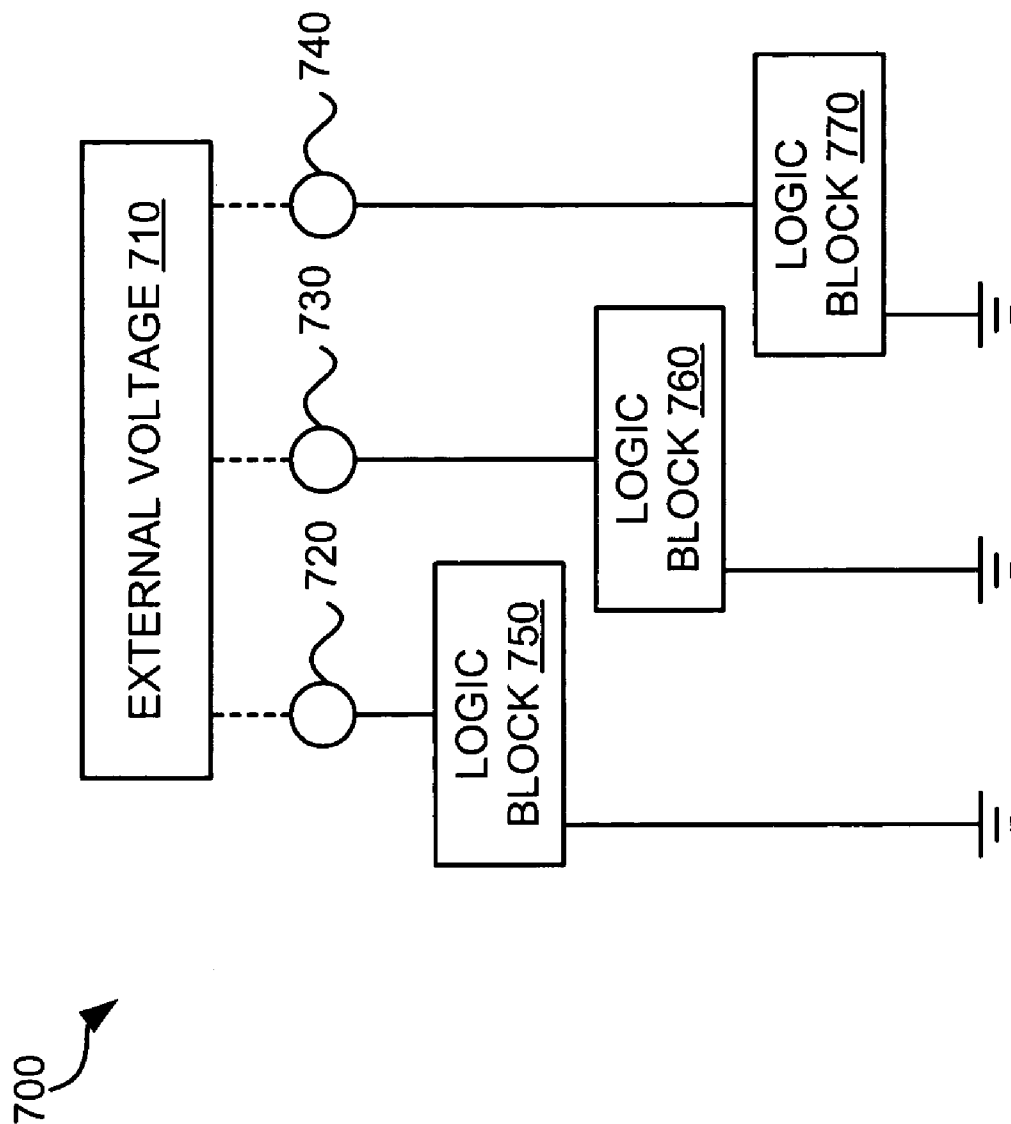


FIG. 7

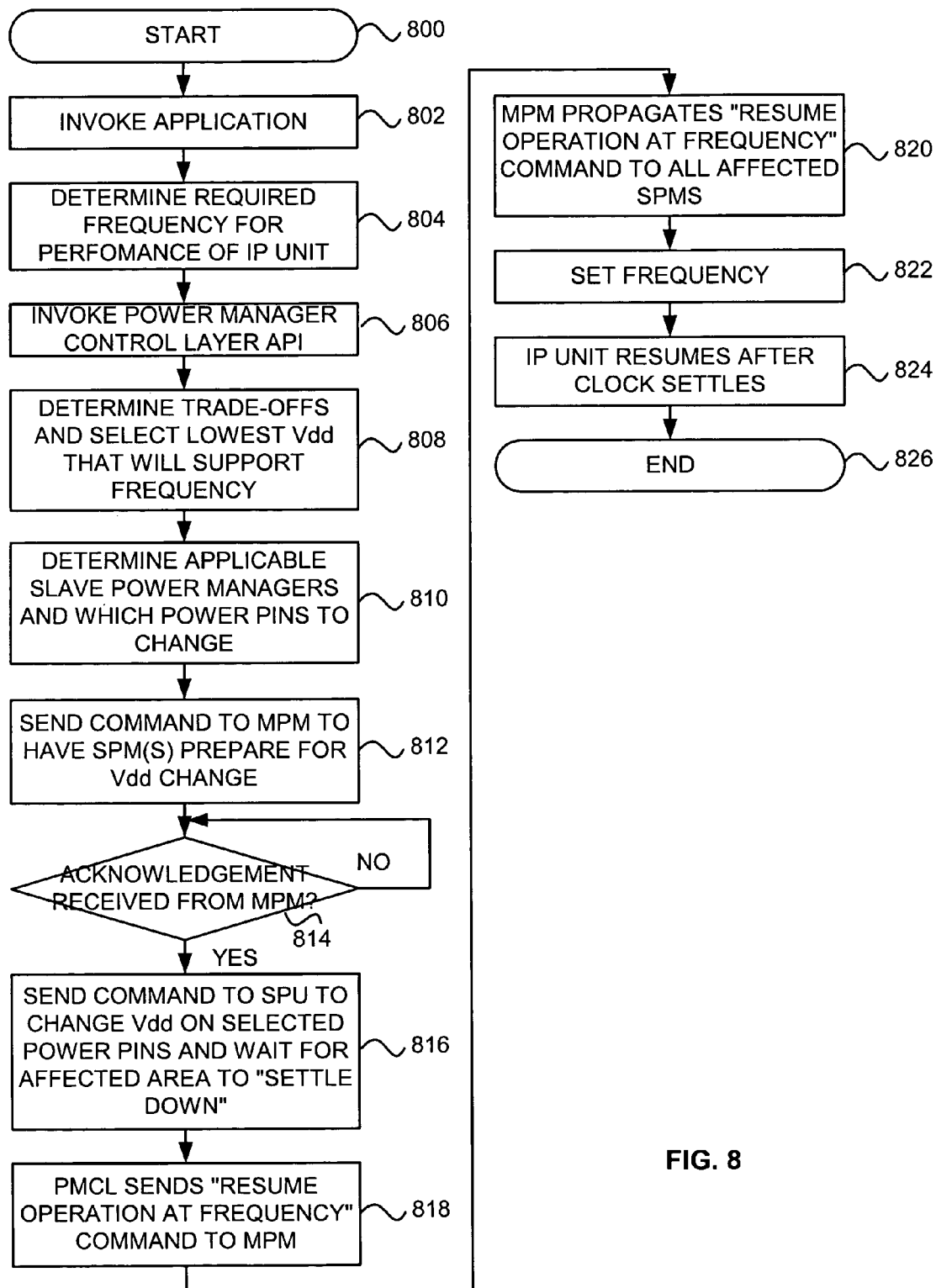


FIG. 8

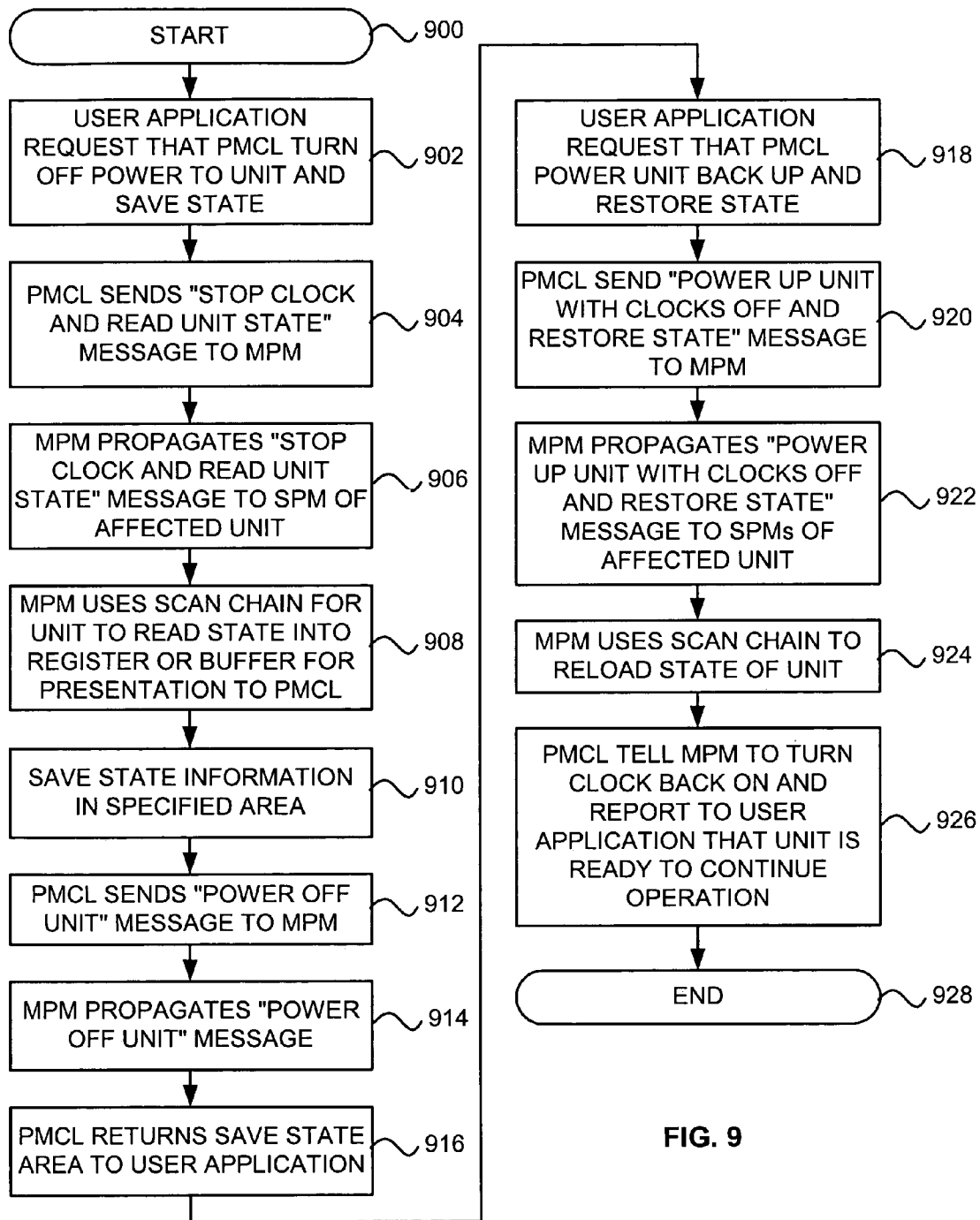


FIG. 9

U.S. Patent

May 23, 2006

Sheet 10 of 13

US 7,051,306 B2

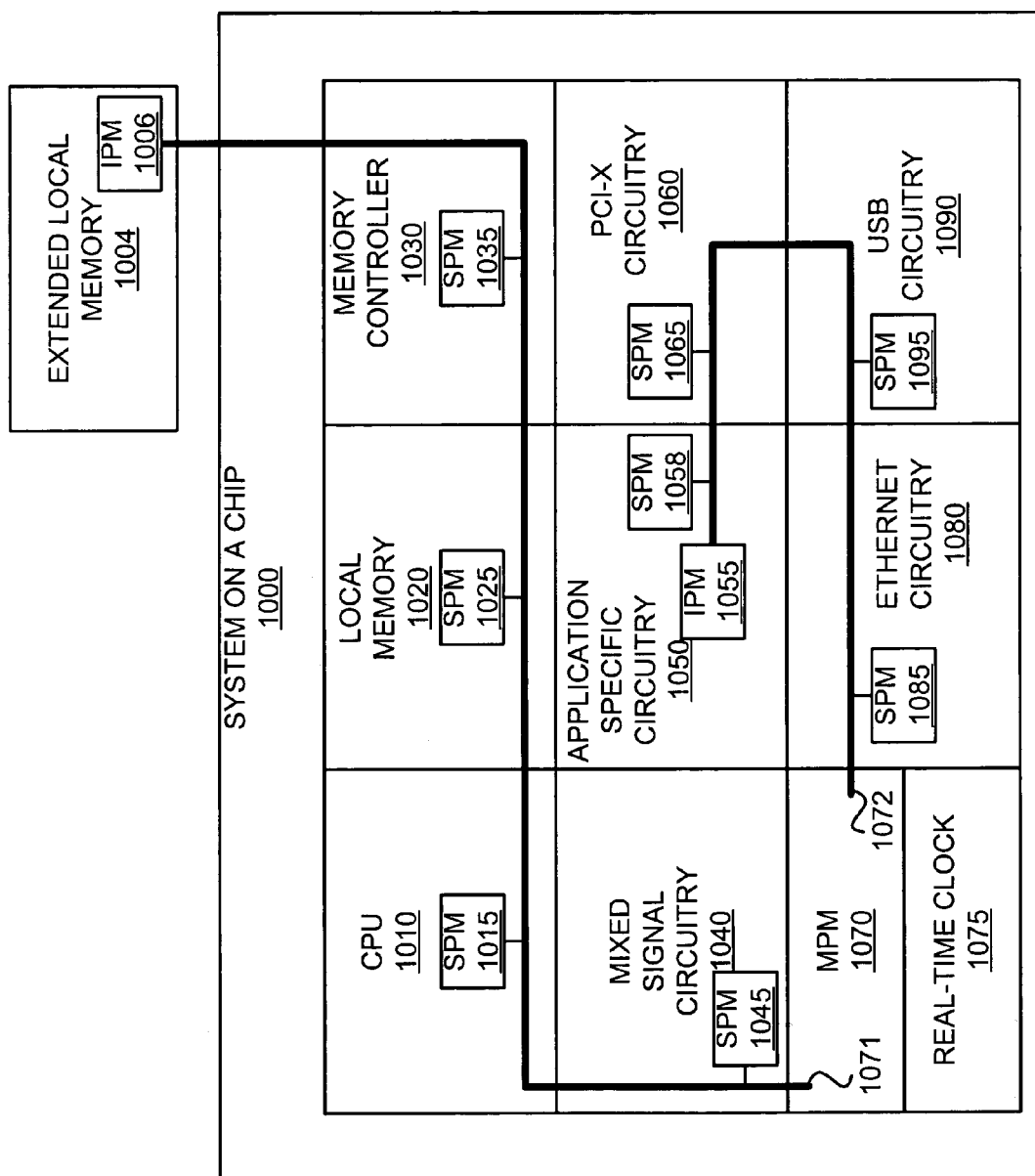


FIG. 10

U.S. Patent

May 23, 2006

Sheet 11 of 13

US 7,051,306 B2

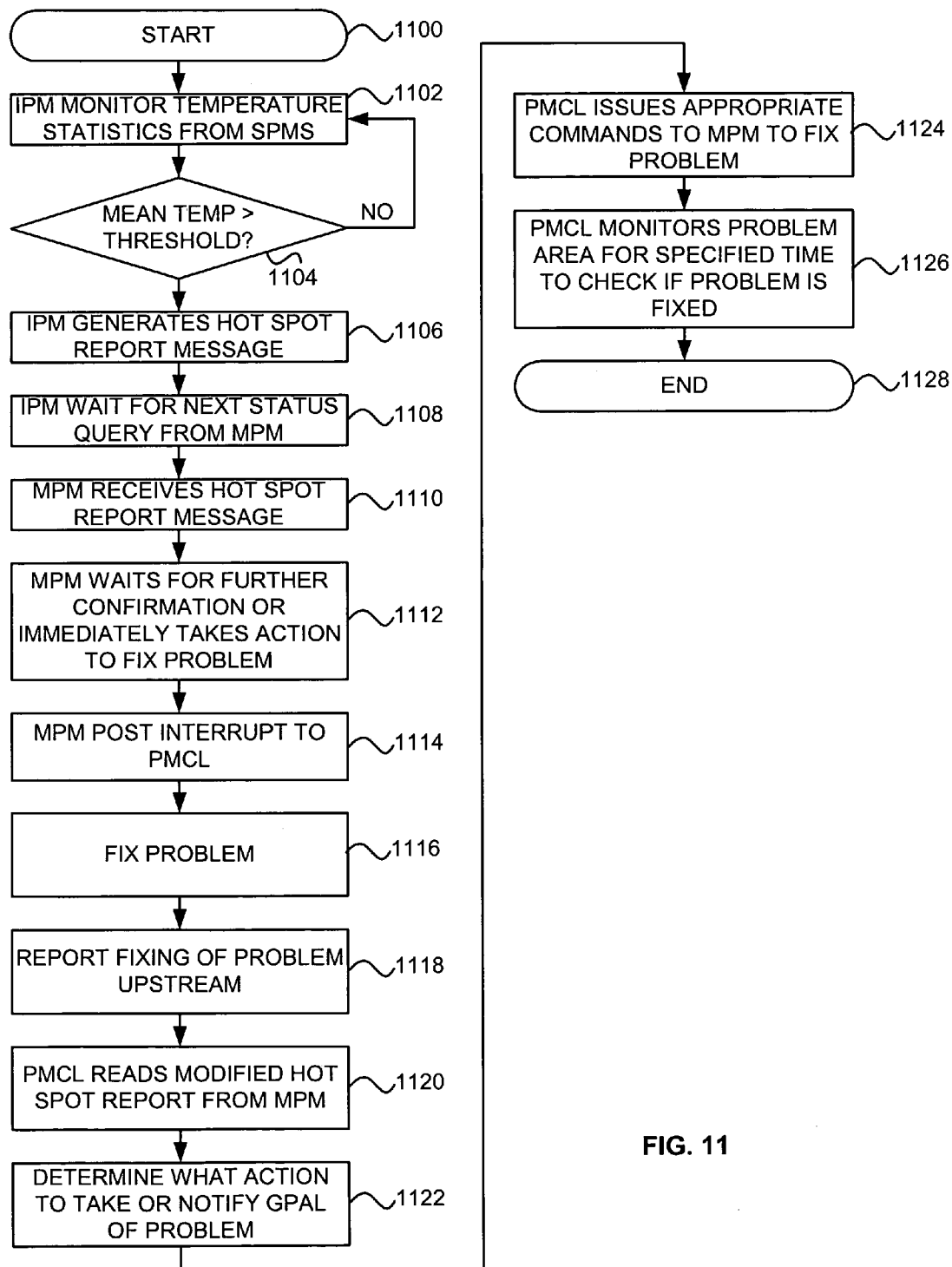


FIG. 11

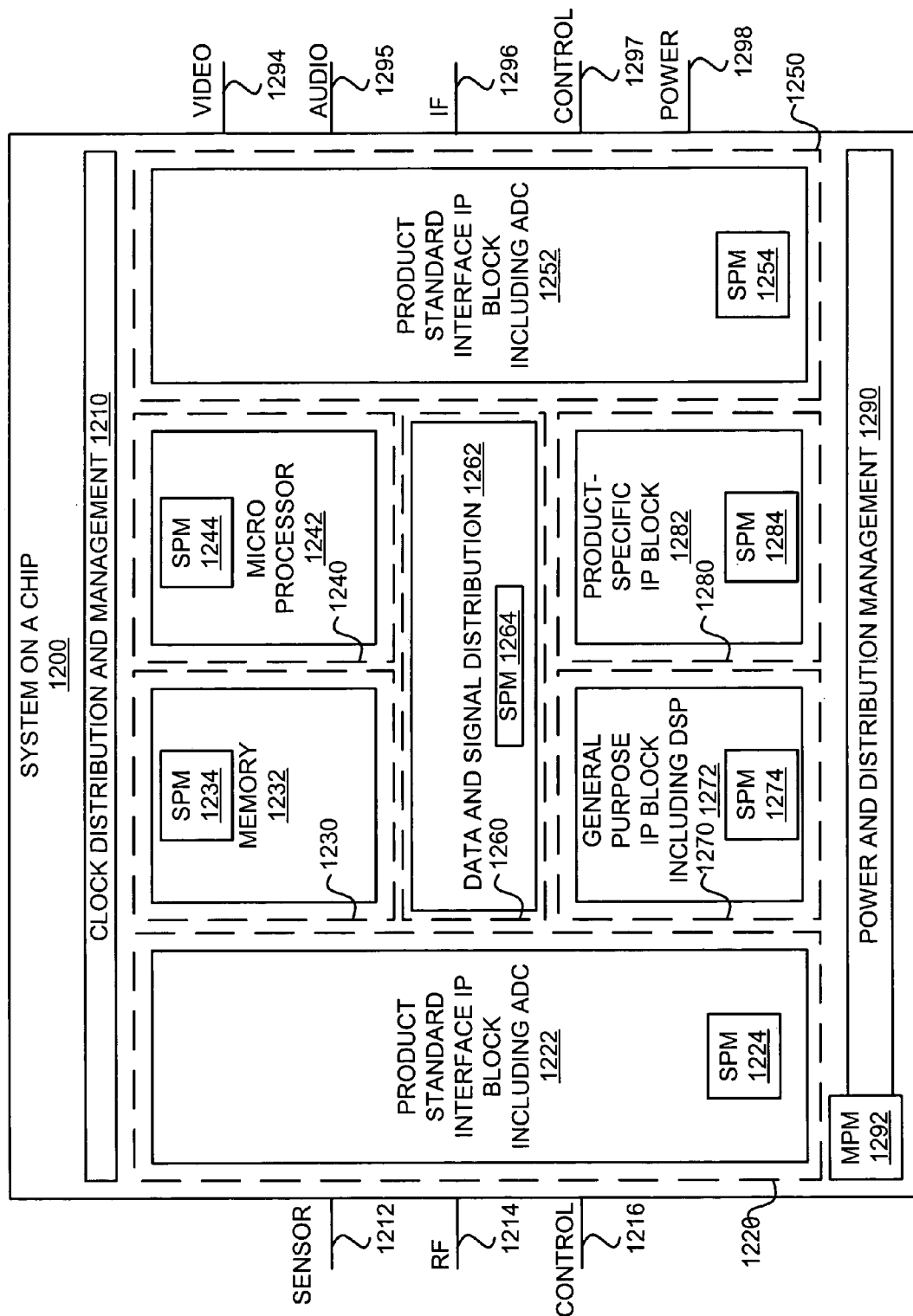


FIG. 12

U.S. Patent

May 23, 2006

Sheet 13 of 13

US 7,051,306 B2

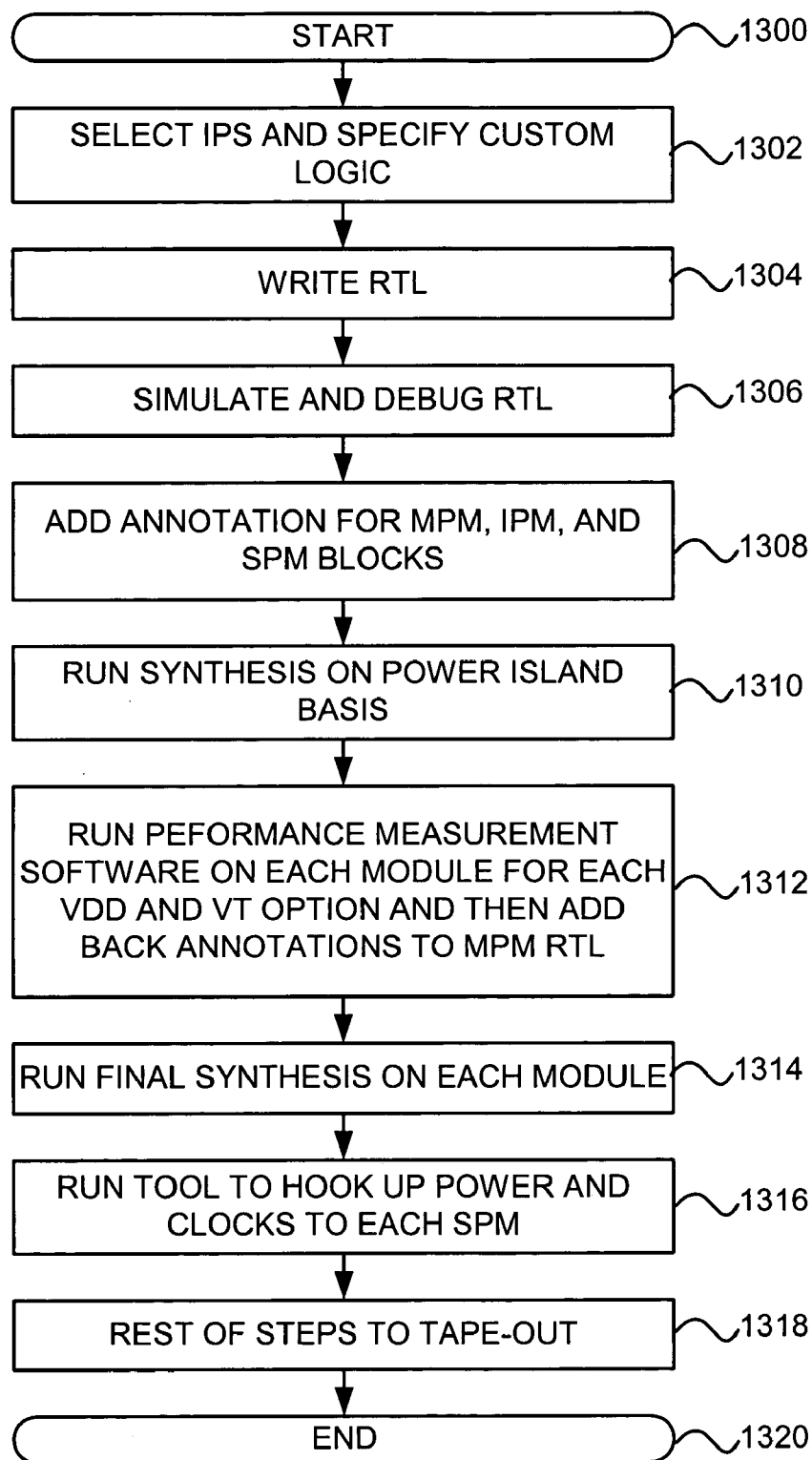


FIG. 13

US 7,051,306 B2

1

**MANAGING POWER ON INTEGRATED
CIRCUITS USING POWER ISLANDS****CROSS-REFERENCE TO RELATED
APPLICATIONS**

This application claims the benefit of U.S. Provisional Application Ser. No. 60/468,742 titled "System and Method for Managing Power in Integrated Circuits," filed May 7, 2003, which is hereby incorporated by reference.

BACKGROUND**1. Field of the Invention**

The present invention relates generally to integrated circuits, and more particularly to managing power on integrated circuits using power islands.

2. Description of the Prior Art

One design goal for integrated circuits is to reduce power consumption. Devices with batteries such as cell phones and laptops especially need a reduction in power consumption in the integrated circuit to extend the charge of the battery. Additionally, a reduction in power consumption prevents overheating and lowers the heat dissipation of the integrated circuit, which in some cases, eliminates or simplifies the heat sinks and/or fans for the integrated circuit.

Some integrated circuits are designed using building blocks of library cells. These library cells are blocks of circuitry performing a function. Some examples of library cells are NAND gates, multiplexers, decoders, comparators, and memory.

In a "full-custom" flow, the integrated circuit is designed at the lowest level such as at the individual transistors, capacitors, and resistors level. The "full-custom" flow may use library cells that are internally developed. The integrated circuit can have optimal performance because the integrated circuit is designed in great detail at the lowest level. However, some problems with the "full-custom" flow are the long time and expensive costs associated with designing at such a detailed level. Furthermore, the "full-custom" flow is cumbersome because the design is at the lowest level.

In a "standard-cell" flow, the integrated circuit is designed using library cells acquired from a third party or other outside source. These library cells are standardized at the logic or function level. The design time for the standard-cell flow is reduced because the library cells are already pre-designed and pre-tested.

In one example for designing an integrated circuit, the library cells are selected, and the custom logic is specified to build the integrated circuit. The register transfer level (RTL) for the integrated circuit is then written for simulation and debugging. After simulation and debugging, the synthesis is run for the integrated circuit. Performance measurement software is executed to determine performance of the integrated circuit. The final synthesis of the integrated circuit can then be run based on the optimal performance of the integrated circuit.

One problem with many integrated circuits is that power consumption is not efficiently utilized. For example, the entire integrated circuit may operate at a maximum frequency just to support an application needing that maximum frequency, while other portions of the integrated circuit can operate at a lower frequency. In another example, inactive circuitry within the integrated circuit consumes power and increases the likelihood of leakage. Inefficient power consumption may also adversely affect performance of the integrated circuit.

2

With the growing complexity of integrated circuits, the reduction of power consumption is even more important when the integrated circuit employs more functionality. One example of an integrated circuit is a system-on-a-chip that includes a microprocessor, memory, I/O interfaces, and an analog-to-digital converter all in a single chip. With many different types of functionality employed within a single chip, the system-on-a-chip consumes even more power than single function integrated circuits.

Some prior integrated circuits have employed voltage islands or multiple clocks to lower power consumption. One problem with these integrated circuits is that the voltages in the power island and the frequencies of the multiple clocks are static. The voltages and the frequencies do not dynamically change based on the needs and operation of the integrated circuit.

SUMMARY OF THE INVENTION

The invention addresses the above problems by managing power in an integrated circuit using power islands. The integrated circuit includes a plurality of power islands where power consumption is independently controlled within each of the power islands. A power manager determines a target power level for one of the power islands. The power manager then determines an action to change a consumption power level of the one of the power islands to the target power level. The power manager performs the action to change the consumption power level of the one of the power islands to the target power level. Power control circuitry controls the power of the one of the power islands.

The power islands may be delineated based on geographic factors or functional circuitry of the integrated circuit. In some embodiments, the action is selecting a clock frequency for one of the power islands or selecting a clock for one of the power islands. In some embodiments, the action is modifying voltage for one of the power islands. The action may be powering on or off one of the power islands.

In some embodiments, the power manager monitors the power consumption level of one of the power islands, determines whether a threshold level was crossed based on the power consumption level, and performs the action based on the crossing of the threshold level. In some embodiments, the power manager saves and restores a state of components within one of the power islands.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of a system for managing power in an integrated circuit in an exemplary implementation of the invention;

FIG. 2 is a diagram of a system for managing power in an integrated circuit in an exemplary implementation of the invention;

FIG. 3 is an illustration for low leakage circuitry for a low power standard cell logic block in an exemplary implementation of the invention;

FIG. 4 is a diagram of a power island in an exemplary implementation of the invention;

FIG. 5 is a flowchart from an application request to a slave power manager action in an exemplary implementation of the invention;

FIG. 6 is an illustration of internal circuitry for a smart power unit in an exemplary implementation of the invention;

FIG. 7 is an illustration of external circuitry for a smart power unit in an exemplary implementation of the invention;

US 7,051,306 B2

3

FIG. 8 is a flowchart from an application request to a smart power unit action in an exemplary implementation of the invention;

FIG. 9 is a flowchart for saving an IP unit's state prior to powering down and restoring the IP unit's state when powering up in an exemplary implementation of the invention;

FIG. 10 is diagram of a system on a chip and an extended local memory in an exemplary implementation of the invention;

FIG. 11 is a flowchart for generating a "hot spot" report from an intermediate power manager to a power management control layer in an exemplary implementation of the invention.

FIG. 12 is a diagram of a system on a chip in an exemplary implementation of the invention; and

FIG. 13 is a flowchart for building a chip with power islands in an exemplary implementation of the invention.

DETAILED DESCRIPTION OF THE INVENTION

As shown in the exemplary drawings wherein like reference numerals indicate like or corresponding elements among the figures, exemplary embodiments of a system and method according to the present invention are described below in detail. It is to be understood, however, that the present invention may be embodied in various forms. Therefore, specific details disclosed herein are not to be interpreted as limiting, but rather as a basis for the claims and as a representative basis for teaching one skilled in the art to employ the present invention in virtually any appropriately detailed system, structure, method, process or manner.

FIG. 1 depicts a block diagram of a system 100 for managing power in an integrated circuit 110 in an exemplary implementation of the invention. The system 100 includes an integrated circuit 110 and a power manager 120. The integrated circuit 110 is any electronic device that is instantiated into silicon and/or related manufacturing materials. One example of the integrated circuit 110 is a system-on-a-chip. The integrated circuit 110 includes multiple IP units, which are blocks of circuitry performing specific functions.

The integrated circuit 110 includes four power islands 112, 114, 116, and 118. FIG. 1 only depicts four power islands 112, 114, 116, and 118 for the sake of simplicity. Other embodiments of the integrated circuit 110 include a plurality of power islands 112, 114, 116, and 118. The power islands 112, 114, 116, and 118 are coupled to a bus 125.

A power island 112, 114, 116, and 118 is any section, delineation, partition, or division of the integrated circuit 110 where power consumption is controlled within the section, delineation, partition, or division. In some embodiments, the power islands 112, 114, 116, and 118 are delineated based on geographic factors of the integrated circuit 110. In some embodiments, the power islands 112, 114, 116, and 118 are delineated based on functional IP units of the integrated circuit 110. In one example depicted in FIG. 10, power islands are delineated by memory, a microprocessor, and separate IP blocks. In some embodiments, the power islands 112, 114, 116, and 118 are asynchronous or synchronous to each other. In some embodiments, the power islands 112, 114, 116, and 118 comprise sub-islands of power to provide further specificity in controlling power in the integrated circuit 110. In some embodiments, each of the power islands 112, 114, 116, and 118 supports multiple clock

4

domains with its own control. In some embodiments, the clocks within the power islands 112, 114, 116, and 118 are variable.

In some embodiments, each of the power islands 112, 114, 116, and 118 includes power control circuitry. Power control circuitry is any circuitry configured to control power within one of the power islands 112, 114, 116, and 118. Some examples of power control circuitry include circuitry for level shifting, signal isolation, Vdd multiplexing, clock multiplexing, and dynamic back bias. In some embodiments, the power control circuitry is included in a standard cell library for standard cell design of the integrated circuit 110.

The power manager 120 is also coupled to the bus 125. One example of the bus 125 is a power command bus described in further detail below in FIG. 10. Other embodiments of the system 100 include numerous variations in which the power manager 120 and the power islands 112, 114, 116, and 118 are interconnected. The power manager 120 is any circuitry, device, or system configured to (1) determine a target power level for one of the power islands 112, 114, 116, and 118 where power consumption is independently controlled within each of the power islands 112, 114, 116, and 118, (2) determine action to change a consumption power level of the one of the power islands 112, 114, 116, and 118 to the target power level, and (3) perform the action to change the consumption power level of the one of the power islands 112, 114, 116, and 118 to the target power level. The power manager 120 can dynamically change the power consumption of the power islands 112, 114, 116, and 118 based on the needs and operation of the integrated circuit 110. The target power level is a desired, calculated, or specified power consumption of the power islands 112, 114, 116, and 118. Some examples of the power manager 120 are the slave power manager (SPM), the intermediate power manager (IPM), and the master power manager (MPM), which are described in further detail below. The power manager 120 may be a hierarchy or group of power managers 120. Although FIG. 1 depicts the power manager 120 as being located outside the integrated circuit 110, other embodiments may have the power manager 120 located in the integrated circuit 110. In other embodiments, the power manager 120 may be distributed among multiple power managers that are on or off the integrated circuit 110 or integrated with a CPU.

The action is any instruction, message, process, function, signal, or variable that controls power consumption in the power island 112, 114, 116, and 118. Some examples of actions are clock gating and dynamic clock selection. Another example of an action is modifying a clock frequency of one of the power islands 112, 114, 116, and 118. Another example of an action is to modify voltage of one of the power islands 112, 114, 116, and 118 such as dynamic voltage source, Vdd, selection. Another example of an action is to control dynamic leakage by controlling variable Vt transistors.

FIG. 2 depicts a diagram of a system 200 for managing power in an integrated circuit 250 in an exemplary implementation of the invention. The system 200 includes an embedded firmware stack 210 and an integrated circuit 250. The embedded firmware stack 210 includes software layers that run on a central processing unit (CPU). In some embodiments, the embedded firmware stack 210 may include an application layer 212, an operating system (OS) sub-system 216, a power management control layer (PMCL) 218, a real-time operating system (RTOS) 220, I/O drivers 222, and power manager (PM) firmware 224 for the MPM, IPM and/or SPM.

US 7,051,306 B2

5

The integrated circuit 250 includes a power island 260, a power island 270, and a MPM 280. The power island 260 includes a low power standard cell logic block 262 and a SPM 264. The power island 270 includes a low power memory block 272 and a SPM 274. The low power standard cell logic block 262 and the low power memory block are coupled to an interface 265. The MPM 280 is coupled to the PMCL 218, the SPM 264, and the SPM 274.

In some embodiments, the application layer 212 includes a global power application layer (GPAL) 214. This GPAL 214 may be useful for more complex applications. If the GPAL 214 is present, then all calls to the PMCL 218 application program interfaces (API) are first directed to the GPAL 214. Both the GPAL 214 and the PMCL 218 provide power management for the integrated circuit 250. In some embodiments, the GPAL 214 and/or the PMCL 218 monitor what load is on each power island 260 and 270. Also, the GPAL 214 and/or the PMCL 218 may build a history of power levels on the power island 260 and 270 and store the history data in a database. The GPAL 214 and the PMCL 218 may also provide data for on-demand resources interested in the power levels of the power islands 260 and 270.

In some embodiments when the GPAL 214 is present, the GPAL 214 provides guidance to the PMCL 218 in making local decisions for power management of the integrated circuit 250. In some embodiments, the GPAL 214 and the PMCL 218 include complex algorithms for a static schedule. The GPAL 214 and the PMCL 218 provide the facility for use code to interact and control power behavior and parameters including providing a database and statistics.

The PMCL 218 also collects information from the MPM 280 and provides commands to the MPM 280 for possible IPMs, SPMs 264 and 274, and the power islands 260 and 270. Some of the commands may be to power on/off, change the power level, or change the frequency of the power islands 260 and 270. In some embodiments with a smart power unit (SPU) 290, the GPAL 214 and the PMCL 218 are responsible for communication with the SPU 290 to implement a power policy and collect information on the actual power supply. The SPU 290 is described in further detail below in FIGS. 6-8. In some embodiments, the GPAL 214 and the PMCL 218 provides sockets for existing power management techniques.

The power manager firmware 224 is firmware executed by the master power manager 280, the intermediate power manager, and the slave power managers 264 and 274.

The MPM 280 is any circuitry configured to control the "global" power behavior of the entire integrated circuit 250. In some embodiments, there are multiple MPMs 280 that control the power behavior of the entire integrated circuit 250. In some embodiments, the MPM 280 communicates with existing IPMs and SPMs 264 and 274 to control power within the power islands 260 and 270. The MPM 280 may receive status information about the power islands 260 and 270 from the SPMs 264 and 274 and the IPMs. The MPM 280 may also determine power trade-offs between the IPMs and the SPMs 264 and 274.

The MPM 280 may also provide a main interface to the PMCL 218 firmware. In some embodiments, the MPM 280 communicates with the PMCL 218 to accept commands (e.g. memory mapped commands) and provide status information about the power within the integrated circuit 250. In some embodiments, the MPM 280 communicates with the PMCL 218 through the main bus for the integrated circuit 250. The MPM 280 may also register each of the SPMs 264 and 274 and IPMs and capabilities with the PMCL 218. Some operations of the MPM 280 are described in further detail

6

below in FIGS. 5, 8, 9, and 11. In some embodiments, the MPM 280 reads and writes scan chains for state save and restore along with the local logic analyzer ability.

Some embodiments may include an IPM (not shown) that coordinates SPMs 264 and 274 on behalf of the MPM 280. In some embodiments, the IPM controls and coordinates power behavior on portions of the area controlled by the MPM 280. In some embodiments, the IPM controls the SPMs 264 and 274 that are on a separate chip from the MPM 280.

The SPM 264 is any circuitry configured to control power within a power island 264 in the integrated circuit 250. One example of the SPM 264 is an IP block that controls power within a power island 264 in the integrated circuit 250. The SPM 264 may include signal buffering, level shifting, and signal isolation. In some embodiments, the SPM 264 is integrated into scan chains to provide easier implementation and integration. Also, in some embodiments, the SPM 264 has a very small "footprint" with a low gate count and low power. In some embodiments, the SPM 264 comprises a command interface to communicate status information, service requests, and commands. Some of the commands are load, sleep, and idle. The SPM 264 may also recognize its own address to allow for multi-drop bussing.

In some embodiments, the SPM 264 has registration capabilities. For registration at power-up, the SPM 264 uses callback registration that is typically used for off chip SPMs and IPMs. For static registration when the integrated circuit is generated for on chip SPMs, the SPM 264 registers what the SPM 264 can do, what the IPM can do, and the type of commands that the SPM 264 or IPM can service. The SPM 264 may also have save and restore functions for power off periods. The SPM 264 may have watch-dog timer(s). In some embodiments, the SPM 264 has a debug interface coordination to the power island 260. The SPM 264 may also monitor local state and collect information for components within the power island 260. The SPM 264 may also locally control Vdd, clocks for frequency selection, and dynamic back biasing. The description for the SPM 264 also applies to the SPM 274 for the power island 270.

In some embodiments, the SPM 264 checks whether an event occurs when the collected information from the power island 260 crosses a threshold or exceeds a range. In some embodiments, the SPM 264, IPMs, or the MPM 280 monitors power consumption levels of the power island 260 to check whether the power consumption levels cross a threshold level or exceed a range. A power consumption level is any information, data, or statistic that indicate the power consumed in a power island 260. Some examples of power consumption levels are temperature and power. The threshold or range may be programmable. The SPM 264 may report the event asynchronously, or another element such as the MPM 280 may solicit whether an event has occurred. The event may also be a multi-level test such as the conditions occur at a greater than a given, programmable frequency or exceed a given, programmable duration.

In some embodiments, the low power standard cell logic block 262 one example of a standard cell library that may be included in the power island 260. In some embodiments, the standard cell library is optimized for lower power. The standard cell library may be characterized over a range of operating voltages. In some embodiments, the standard cell library includes synchronous circuits and/or asynchronous circuits. In some embodiments, the standard cell library includes static circuits and/or encapsulated, dynamic logic circuits. The standard cell library may also include multiple voltage domain interface circuits such as level shifters and

US 7,051,306 B2

7

signal isolation circuits. The standard cell library may also have a multi-threshold design and characterization such as standard V_t , high V_t , low V_t , and variable V_t circuits. The standard cell library may also include data retention (shadow) circuits and anti-glitch circuits. The standard cell library may also include low leakage “sleep” circuits. FIG. 3 depicts an illustration of low leakage circuitry 300 for the low power standard cell logic block 262 in an exemplary implementation of the invention.

FIG. 4 depicts a diagram of the power island 270 in an exemplary implementation of the invention. The power island 270 includes a low power memory block 272 and the SPM 274. The low power memory block 272 includes banked architectures 410, 420, 430, and 440 for memories and sleep, power down circuitry 450. In some embodiments, the memory in the power island 270 is RAM and/or ROM. Some examples of the RAM are the SRAM compilers such as single-port, 2-port, and dual-port. Some examples of the ROM are ROM compilers. Some memories in the power island 270 are optimized for low power such as the low power memory block 272. In one embodiment, the low power memory block 272 includes multiple bank architectures via the compiler such as banked architectures 410, 420, 430, and 440. The memories in the power island 270 may also include sleep, power down circuitry 450 for low power modes such as sleep, nap, and full power down. The memories in the power island 270 may also include programmable read/write ports. The memories in the power island 270 may also be an asynchronous and/or synchronous design.

In some embodiments, the system 200 also includes a smart power unit (SPU) 290. In some embodiments, the SPU 290 is off-chip to the integrated circuit 250. The SPU 290 is an external unit configured to control power and clock distribution to the integrated circuit. The circuitry of the SPU is described in further detail below in FIGS. 7 and 8.

FIG. 5 depicts a flowchart from an application request to an SPM action in an exemplary implementation of the invention. FIG. 5 begins in step 500. In step 502, an application is invoked such as playing a movie. In step 504, the application determines the required frequency for performance of an IP unit. For example, the application determines a specified frequency in MHz for IP unit for an MPEG decoder. In some embodiments, when the IP unit is totally asynchronous, the required frequency will be an artificial measure of performance. In some embodiments that have multiple clocks involved, the minimum performance for each clock is specified.

In step 506, the application invokes the PMCL 218 API call. One example of this call is “Set_Rate (unit Y, N MHz, degree of dynamic power management (DPM) allowed, DPM threshold, other information to allow DPM trade-offs, starting back bias for SPMs with dynamic bias, wait for power up flag).” In this example, the other information could be “no power off, use high V_t and clocks off instead” and “major waits are about 10 us, with a 400 ns threshold.” In some embodiments with multiple clocks, the PMCL 218 API call allows the application to specify all required frequencies.

The PMCL 218 determines possible trade-offs available for the unit and selects a frequency rate, V_{dd} , and V_t if applicable for the unit that best meets the given requirements in step 508 and determines the applicable SPMs 264 and 274 in step 510. In some embodiments, the MPM 280 or IPM performs steps 508 and 510. In some embodiments for

8

multiple clocks, the V_{dd} and V_t specified would allow for all specified clocks to meet or exceed their required frequencies.

In step 512, the PMCL 218 then writes the desired setting for the SPMs 264 and 274 to the MPM 280 (or IPM). In step 514, the MPM 280 (or IPM) converts the request into one or more commands for the SPMs 264 and 274 (or IPM) associated with the unit.

In step 516, if the application sets the wait for power up flag, the PMCL 218 then waits until the IP unit is fully powered up before returning from the call. Otherwise, the call returns as soon as the command is acknowledged, with status of 0 for okay, 1 for fast power-up, 2 for slow power-up, or 3+ for error conditions. The powered up state is when the unit is at the desired V_{dd} for the requested frequency, and not just that the unit is turned on.

In step 518, the MPM 280 (or IPM) sends the requests to the appropriate targets. In step 520, the MPM 280 waits for the receipt of the acknowledgements that indicate message received and executed or execution has begun. A NACK or negative acknowledgement may also be returned from the SPM 264 and 274.

In step 522, the SPM 264 receives the commands and performs the actions. Steps 524–528 are possible actions that the SPM 264 may perform. In step 524, the SPM 264 switches the V_{dd} mux. In step 526, the SPM 264 switches a clock mux. In some embodiments when the voltage goes down, step 526 is performed before step 524. In step 528, the SPM 264 changes the V_t on the associated transistors. After the SPM 264 performs the actions, the SPM 264 returns a status message upstream indicating acknowledgement or negative acknowledgement in step 530. FIG. 5 ends in step 532.

FIG. 6 depicts an illustration of internal circuitry 600 for an SPU 290 in an exemplary implementation of the invention. In this embodiment, the internal circuitry 600 for the SPU 290 is internal to the integrated circuit 250 of FIG. 2. The internal circuitry 600 includes an external voltage 610, a dropout voltage regulator 620, a dropout voltage regulator 630, a logic block 640, and a logic block 650.

FIG. 7 depicts an illustration of external circuitry 700 for an SPU 290 in an exemplary implementation of the invention. In this embodiment, the external circuitry 700 for the SPU 290 is external to the integrated circuit 250 of FIG. 2. The internal circuitry 700 includes an external voltage 710, a power supply pin 720, a power supply pin 730, a power supply pin 740, a logic block 750, a logic block 760, and a logic block 770. The external circuitry 700 for the SPU 290 provides a DC/DC conversion. The DC/DC conversion provides multiple independent power supply pins 720, 730, and 740. The power supply pins 720, 730, and 740 have a variable voltage supply on each separate power pin. Also, in some embodiments, the variable voltage is within a range and in steps. In some embodiments, the PMCL 218 controls the voltage to power supply pins 720, 730, and 740.

FIG. 8 depicts a flowchart from an application request to an SPU action in an exemplary implementation of the invention. FIG. 8 begins in step 800. In step 802, an application is invoked. In step 804, the application determines the required frequency for performance of an IP unit. For example, the application determines a specified frequency in MHz for IP unit for an MPEG decoder.

In step 806, the application invokes the PMCL 218 API call. One example of this call is “Set_Rate (unit Y, N MHz, degree of dynamic power management (DPM) allowed, DPM threshold, other information to allow DPM trade-offs, starting back bias for SPMs with dynamic bias, wait for

US 7,051,306 B2

9

power up flag).” In this example, the other information could be “no power off, use high Vt and clocks off instead” and “major waits are about 10 us, with a 400 ns threshold.”

The PMCL 218 determines possible trade-offs available for the unit and selects a the lowest possible Vdd that will support the requested frequency in step 808 and determines the SPMs 264 and 274 that will be affected and which power PINs to change in step 810.

In step 812, the PMCL 218 sends a command to the MPM 280 to have the SPMs 264 and 274 (and IPM) to prepare for a Vdd change. In step 814, the PMCL 218 waits for the acknowledgement from the MPM 280. In step 816, the PMCL 218 sends a command to the SPU 290 to change the Vdd on the selected power pins and waits for the affected area to “settle down.” In step 818, the PMCL 218 then sends a “resume operation at a specified frequency” command to the MPM 280. In step 820, the MPM 280 propagates the resume command to all affected SPMs 264 and 274 (and IPMs). In step 822, one of the power managers (i.e. MPM 280, IPM, or SPMs 264 and 274) sets the specified frequency. In step 824, the IP unit operation resumes after the clock has settled. FIG. 8 ends in step 826. In some embodiments, the user application has the option of waiting for the entire operation to finish or to continue and either query the PMCL 218 as to the progress of the operation or wait for a “finished” interrupt from the PMCL 218.

FIG. 9 depicts a flowchart for saving an IP unit’s state prior to powering down and restoring the IP unit’s state when powering up in an exemplary implementation of the invention. FIG. 9 begins in step 900. In step 902, the user application requests that the PMCL 218 power off an IP unit and wants to save the IP unit’s state. In some embodiments, the reconfiguration of the IP unit takes a long time. The request in step 902 may be accompanied by the address of an area to which the state should be saved.

In step 904, the PMCL 218 sends a “stop clock and read IP unit state” message to the MPM 280. The MPM 280 then propagates the “stop clock and read IP unit state” message to the SPMs 264 and 274 of the affected unit in step 906. In step 908, the MPM 280 uses the scan chain for the IP unit to read the state into a register or buffer for presentation to the PMCL 218. In step 910, if the PMCL 218 provided the MPM 280 with the address of the save area, the MPM 280 saves the state information directly in the specified area. In step 912, after all of the IP units’ state has been saved, the PMCL 218 sends a “power off IP unit” message to the MPM 280. In step 914, the MPM 280 then propagates out the “power off IP unit” message. In step 916, the PMCL 218 returns the save state area to the user application. The save state area contains the state of the unit.

At a later time, in step 918, the user application requests that the PMCL 218 power the IP unit back up and restore the state of the IP unit. In some embodiments, the user application request includes the address of the area to which the state was saved. In step 920, the PMCL 218 sends a “power up IP unit with clocks off and restore state” message to the MPM 280. In step 922, the MPM 280 propagates the “power up IP unit with clocks off and restore state” message to the SPMs 264 and 274 of the affected IP unit. In step 924, after the IP unit is powered up, the MPM 280 uses the scan chain to reload the unit’s state. In some embodiments, the reloading of the unit’s state originates directly from the save area or from information passed to the MPM 280 from the PMCL 218. In step 926, the PMCL 218 sends a message to the MPM 280 to turn clock back on and report to user application that the IP unit is ready to continue operations. FIG. 9 ends in step 928.

10

In some embodiments, the same functionality of FIG. 9 can be used to implement an internal “logic analyzer” function, in which the IP unit in question would not be powered down after being read. If the associated SPMs 264 and 274 of the IP unit have the ability to single- or multi-step the clock, the local scan testing is performed by having the SPMs 264 and 274 “signal isolate” the IP unit. Then, use of a combination of the ability of single- or multi-step the clock and the ability to use the scan chain can read/write the IP unit’s internal state.

FIG. 10 depicts a diagram of a system-on-a-chip (SOC) 1000 and an extended local memory 1004 in an exemplary implementation of the invention. The SOC 1000 is one example of the integrated circuit 250 and communicates with the embedded firmware stack 210 in FIG. 2 as described above. The SOC 1000 includes a CPU 1010, a local memory 1020, a memory controller 1030, a mixed signal circuitry 1040, an application specific circuitry 1050, a PCI-X circuitry 1060, a MPM 1070, a real-time clock (RTC) 1075, an Ethernet circuitry 1080, and a USB circuitry 1090. The CPU 1010, the local memory 1020, the memory controller 1030, the mixed signal circuitry 1040, the application specific circuitry 1050, the PCI-X circuitry 1060, the Ethernet circuitry 1080, and the USB circuitry 1090 are all power islands in which power is controlled within the power islands by power managers. In this embodiment, the power islands are delineated by the functionality of a part of the SOC 1000.

The extended local memory 1004 includes an IPM 1006 that is coupled to the bus 1071. The CPU 1010 includes an SPM 1015 that is coupled to the bus 1071. The local memory 1020 includes an SPM 1025 that is also coupled to the bus 1071. The memory controller 1030 comprises an SPM 1035 that is coupled to the bus 1071. The mixed signal circuitry 1040 includes an SPM 1045 that is coupled to the bus 1071. The application specific circuitry 1050 includes an IPM 1055 and an SPM 1058 that are both coupled to the bus 1072. The PCI-X circuitry 1060 includes an SPM 1065 that is coupled to the bus 1072. The MPM 1070 is coupled to the bus 1071 and the bus 1072. The MPM 1070 is one example of the MPM 280 as described above. The Ethernet circuitry 1080 includes an SPM 1085 that is coupled to the bus 1072. The USB circuitry 1090 includes an SPM 1095 that is coupled to the bus 1072.

In this embodiment, power command buses comprise the bus 1071 and the bus 1072. The bus 1071 and 1072 are simple multi-drop serial buses that may cross chip boundaries and interconnect power managers. The power command bus may be a combination of serial busses, such as the bus 1071 and the bus 1072, with one per region of the chip and then multi-drop within the region. In other embodiments, the power command bus includes a parallel bus or a combination of serial and parallel busses. In some embodiments, the power command bus is the system bus. The power command bus may contain messages with at least a unit ID with the associated payload. For a fixed point to point bus, the message does not need a unit ID only the payload.

In some embodiments, the power command bus uses an error detection scheme such as parity, ECC, or a redundant code. In some embodiments, the power command bus is a low performance bus that does not interfere with the integrated circuit design and is not visible to the user. In some embodiments, the communication between the PMCL 218 and the MPM 1070 is memory mapped and based on the main bus such as an AHB for an SOC 1000.

US 7,051,306 B2

11

Some embodiments may include a separate bus for reporting status information between the power managers. In one example, this separate bus provides asynchronous “Alert” type status messages from SPMs to the MPM.

There can be various message formats for the power command bus. In one example for a basic format, the message format includes a start of message indicator, a power manager address, type code, basic command, and an end of message indicator. In one example for an extended format, the message format includes the basic format, an additional length, and additional information. In one example for a response message format, the message format includes a start of message reply indicator, a power manager address, a 3b ACK or NAK or return status (implied ACK), a payload length for return status messages, a payload for return status messages, a reason code for NAK, and an end of message indicator.

In some embodiments, the SPM 1015 or the IPM 1006 times out on replying to a command, and the MPM 1070 reissues the command a programmable number of times. If the SPM 1015 or the IPM 1006 still fails, the MPM 1070 marks the SPM 1015 or the IPM 1006 as unusable and reports back the failure to the PMCL 218. In one embodiment, the PMCL 218 has the MPM 1070 reinitialize the failing SPM 1015 or the IPM 1006 via the scan system and then retries sending the message. Other conditions such as invalid responses to commands or invalid status reports are also handled by the re-initialization. In some embodiments, the state of the failing SPM 1015 or the IPM 1006 may be read out and saved for later analysis.

FIG. 11 depicts a flowchart for generating a “hot spot” report from an IPM 1055 to the PMCL 218 in an exemplary implementation of the invention. FIG. 11 begins in step 1100. In step 1102, the IPM 1055 monitors the temperature statistics of the SPMs. In step 1104, the IPM 1055 checks whether the mean temperature has exceeded a predetermined and programmed threshold. If the mean temperature has not exceeded the threshold, the IPM 1055 returns to step 1102 to keep monitoring. If the mean temperature has exceeded the threshold, the IPM 1055 then generates a problem (“hot spot”) report message in step 1106. In step 1108, the IPM 1055 waits for the next status query the MPM 1070. In some embodiments, a logically separate bus is provided for asynchronous status reports to the MPM 280.

In step 1110, the MPM 1070 receives the hot spot report message. In step 1112, the MPM 1070 either waits for further confirmation (i.e. see that the “hot spot” last pasts a predetermined threshold) or immediately takes action to fix the problem depending on the internal logic of the MPM 1070. In some embodiments, in step 1114, the MPM 1070 takes action by posting an interrupt to the PMCL 280 that it needs attention. In other embodiments, if the PMCL 280 queries the MPM 1070 often enough, then the interrupt is not needed. In step 1116, the MPM 1070 (or IPM) fixes the problem by making a local trade-off, performing the action, or ordering the action to be performed. One example of fixing the problem is lowering the operating frequency of the hot spot area. In step 1118, the MPM 1070 reports both the problem and the fixing of the problem upstream.

In step 1120, the PMCL 218 reads the modified hot spot report from the MPM 1070. In step 1122, the PMCL 218 determines what action to take to fix the problem or notify the GPAL 214 of the problem. In step 1124, the PMCL 218 issues the appropriate commands to the MPM 1070 to fix the problem. In this step, the software will make the trade-off required to fix the problem. In some embodiments, if the GPAL 214 determines the high level fix, then the GPAL 214

12

sends it to the PMCL 280 to be converted into MPM commands. In step 1126, the PMCL 218 monitors the problem area for a specified time to check if the problem is fixed. FIG. 11 ends in step 1128.

In some embodiments, the type of functionality in FIG. 11 can be performed at other levels such as the PMCL 218. In one example, the PMCL 218 queries (via the MPM 1070) all the SPMs that have the desired measurement ability (and IPMs) as to their local conditions and can then, for example, build a “map” of power usage that it can then act on.

FIG. 12 depicts a diagram of a system-on-a-chip (SOC) 1200 in another exemplary implementation of the invention. The SOC 1200 is attached to a sensor link 1212, an RF link 1214, a control link 1216, a video link 1294, an audio link 1295, an interface link 1296, a control link 1297, and a power link 1298. The SOC 1200 includes clock and distribution management 1210, an IP block power island 1220, a memory power island 1230, a microprocessor power island 1240, an IP block power island 1250, a distribution power island 1260, an IP block power island 1270, an IP block power island 1280, and power and distribution management 1290.

The IP block power island 1220 includes a product standard interface IP block including an analog-to-digital converter (ADC) 1222, which includes an SPM 1224. The IP block power island 1230 includes a memory 1232, which includes an SPM 1234. The microprocessor power island 1240 includes a microprocessor 1242, which includes an SPM 1244. The IP block power island 1250 includes a product standard interface IP block including ADC 1252, which includes an SPM 1254. The distribution power island 1260 includes a data and signal distribution 1262, which includes an SPM 1264. The IP block power island 1270 includes a general purpose IP block including a digital signal processor (DSP) 1272, which includes an SPM 1274. The IP block power island 1280 includes a product-specific IP block 1282, which includes an SPM 1284. The power and distribution management 1290 includes an MPM 1292.

FIG. 13 depicts a flowchart for building a chip with power islands in an exemplary implementation of the invention. FIG. 13 begins in step 1300. In step 1302, the IPs or library cells are selected, and the custom logic, if any, is specified to be used to build the chip. In some embodiments, the maximum and sub clock rate, % idle time, and minimum and maximum Vdd are specified. In step 1304, the register transfer level (RTL) is written. In step 1306, the RTL is simulated and debugged.

Either during or after the writing of the RTL, a software tool may be used to add annotations for the MPM, IPM, and/or SPM blocks in step 1308. In some embodiments, the desired functionality options are specified in the annotations. In some embodiments, the optional functionality is specified per SPM block. In some embodiments, the RTL is annotated with SPM configuration information on a per-module basis. In some embodiments, the proper annotation is inserted by hand on a per module basis during or after the RTL is written. In other embodiments, a separate table in the proper format is created with the module names and respective annotations. In some embodiments, a separate software tool provides the ability to interactively build the separate table. In some embodiments, the software tool adds the proper annotation to the un-annotated modules in the RTL using information from the table.

In step 1310, the synthesis is run on a power island basis, where power islands are not mixed. In step 1312, performance measurement software is run on each module for each Vdd and Vt option, and then the annotations for the

US 7,051,306 B2

13

MPM are added to the RTL. In some embodiments, the performance of the design is measured at all desired voltages and Vt combinations. The software tool then derives the actual frequency, Vdd, and Vt tables for each SPM and back annotates the MPM (or IPM) RTL of the SPM characteristics.

In step 1314, the final synthesis is run on a power island basis. In some embodiments, the modules are separately routed or by SPM. In step 1316, the software tool is run to hook up power and clocks to each SPM. Step 1318 is the rest of the steps to tape-out. In some embodiments, the software tool generates the finished net-list with all information. FIG. 13 ends in step 1320.

The above-described elements can be comprised of instructions that are stored on storage media. The instructions can be retrieved and executed by a processor. Some examples of instructions are software, program code, and firmware. Some examples of storage media are memory devices, tape, disks, integrated circuits, and servers. The instructions are operational when executed by the processor to direct the processor to operate in accord with the invention. Those skilled in the art are familiar with instructions, processor, and storage media.

The above description is illustrative and not restrictive. Many variations of the invention will become apparent to those of skill in the art upon review of this disclosure. The scope of the invention should, therefore, be determined not with reference to the above description, but instead should be determined with reference to the appended claims along with their full scope of equivalents.

What is claimed is:

1. A method of managing power on an integrated circuit including a plurality of power islands, the method comprising:

determining a target power level for one of the power islands based on needs and operation of the integrated circuit where power consumption is independently controlled within each of the power islands;

determining at least one of actions to change a power consumption level of the one of the power islands to the target power level; and

performing the at least one of the actions to change the power consumption level of the one of the power islands to the target power level where one of the actions comprises selecting a frequency for the one of the power islands.

2. The method of claim 1 wherein the power islands are delineated based on geographic factors of the integrated circuit.

3. The method of claim 1 wherein the power islands are delineated based on functional circuitry of the integrated circuit.

4. The method of claim 1 wherein the power islands comprise power sub-islands.

5. The method of claim 1 wherein the power islands comprise multiple clock domains.

6. The method of claim 1 wherein another one of the actions comprises selecting a clock for the one of the power islands.

7. The method of claim 1 wherein another one of the actions comprises modifying a first voltage for the one of the power islands.

8. The method of claim 7 wherein the first voltage comprises a source voltage.

9. The method of claim 7 wherein the first voltage comprises a threshold voltage.

14

10. The method of claim 1 wherein another one of the actions comprises powering on the one of the power islands.

11. The method of claim 1 wherein another one of the actions comprises powering off the one of the power islands.

12. The method of claim 1 wherein another one of the actions comprises changing the one of the power islands to a sleep mode.

13. The method of claim 1 further comprising:

monitoring the power consumption level of the one of the power islands;

determining whether a threshold level was crossed based on the power consumption level; and

performing another one of the based on the crossing of the threshold level.

14. The method of claim 1 further comprising saving a state of components within the one of the power islands.

15. The method of claim 1 further comprising restoring a state of components within the one of the power islands.

16. A system for an integrated circuit comprising a plurality of power islands where power consumption is independently controlled within each of the power islands, the system comprising:

power control circuitry configured to control power for one of the power islands; and

a power manager configured to determine a target power level for the one of the power islands based on needs and operation of the integrated circuit, determine at least one of actions to change a power consumption level of the one of the power islands to the target power level, and perform the at least one of the actions to change the power consumption level of the one of the power islands to the target power level where one of the actions comprises selecting a frequency for the one of the power islands.

17. The system of claim 16 wherein the power islands are delineated based on geographic factors of the integrated circuit.

18. The system of claim 16 wherein the power islands are delineated based on functional circuitry of the integrated circuit.

19. The system of claim 16 wherein the power islands comprise power sub-islands.

20. The system of claim 16 wherein the power islands comprise multiple clock domains.

21. The system of claim 16 wherein another one of the actions comprises selecting a clock for the one of the power islands.

22. The system of claim 16 wherein another one of the actions comprises modifying a first voltage for the one of the power islands.

23. The system of claim 22 wherein the first voltage comprises a source voltage.

24. The system of claim 22 wherein the first voltage comprises a threshold voltage.

25. The system of claim 16 wherein another one of the actions comprises powering on the one of the power islands.

26. The system of claim 16 wherein another one of the actions comprises powering off the one of the power islands.

27. The system of claim 16 wherein another one of the actions comprises changing the one of the power islands to a sleep mode.

28. The system of claim 16 wherein the power manager is configured to monitor the power consumption level of the one of the power islands, determine whether a threshold level was crossed based on the power consumption level, and perform the at least one of the actions based on the crossing of the threshold level.

US 7,051,306 B2

15

29. The system of claim 16 wherein the power manager is configured to save a state of components within the one of the power islands.

30. The system of claim 16 wherein the power manager is configured to restore a state of components within the one of the power islands.

31. The system of claim 16 further comprising a bus coupled to the power manager and the power islands.

32. The system of claim 16 wherein the power manager comprises a master manager configured to control power of the integrated circuit.

33. The system of claim 16 wherein the power manager comprises a slave manager configured to control power of the one of the power islands.

34. The system of claim 16 wherein the power control circuitry includes clock multiplexing circuitry.

35. The system of claim 16 wherein the power control circuitry includes voltage multiplexing circuitry.

36. The system of claim 16 wherein the power control circuitry includes level shifting circuitry.

37. A software product for managing power on an integrated circuit including a plurality of power islands, the software product comprising:

power management software operational when executed by a processor to direct the processor to determine a target power level for one of the power islands based on needs and operation of the integrated circuit where power consumption is independently controlled within each of the power islands, determine at least one of actions to change a power consumption level of the one of the power islands to the target power level, and perform the at least one of the actions to change the power consumption level of the one of the power islands to the target power level where one of the actions comprises selecting a frequency for the one of the power islands; and

a software storage medium operational to store the power management software.

38. The software product of claim 37 wherein the power islands are delineated based on geographic factors of the integrated circuit.

39. The software product of claim 37 wherein the power islands are delineated based on functional circuitry of the integrated circuit.

16

40. The software product of claim 37 wherein the power islands comprise power sub-islands.

41. The software product of claim 37 wherein the power islands comprise multiple clock domains.

42. The software product of claim 37 wherein another one of the actions comprises selecting a clock for the one of the power islands.

43. The software product of claim 37 wherein another one of the actions comprises modifying a first voltage for the one of the power islands.

44. The software product of claim 43 wherein the first voltage comprises a source voltage.

45. The software product of claim 43 wherein the first voltage comprises a threshold voltage.

46. The software product of claim 37 wherein another one of the actions comprises powering on the one of the power islands.

47. The software product of claim 37 wherein another one of the actions comprises powering off the one of the power islands.

48. The software product of claim 37 wherein another one of the actions comprises changing the one of the power islands to a sleep mode.

49. The software product of claim 37 wherein the power management software is operational when executed by the processor to direct the processor to monitor the power consumption level of the one of the power islands, determine whether a threshold level was crossed based on the power consumption level, and perform the at least one of the actions based on the crossing of the threshold level.

50. The software product of claim 37 wherein the power management software is operational when executed by the processor to direct the processor to save a state of components within the one of the power islands.

51. The software product of claim 37 wherein the power management software is operational when executed by the processor to direct the processor to restore a state of components within the one of the power islands.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 7,051,306 B2
APPLICATION NO. : 10/840893
DATED : May 23, 2006
INVENTOR(S) : Barry Alan Hoberman et al.

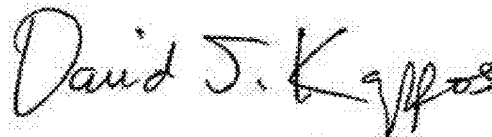
Page 1 of 1

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

In Column 14, Claim 12, line 6, before “comprises” insert --actions--.

In Column 14, Claim 13, line 13, after “one of the” insert --actions--.

Signed and Sealed this
Twenty-first Day of June, 2011

A handwritten signature in black ink that reads "David J. Kappos". The signature is written in a cursive, flowing style.

David J. Kappos
Director of the United States Patent and Trademark Office

EXHIBIT C

US007349448B2

(12) **United States Patent**
Chang(10) **Patent No.:** **US 7,349,448 B2**
(45) **Date of Patent:** **Mar. 25, 2008**(54) **DISTRIBUTED MULTIPLEXING CIRCUIT
WITH BUILT-IN REPEATER**(75) Inventor: **Edward Chang**, Ft. Collins, CO (US)(73) Assignee: **Hewlett-Packard Development
Company, L.P.**, Houston, TX (US)(*) Notice: Subject to any disclaimer, the term of this
patent is extended or adjusted under 35
U.S.C. 154(b) by 1000 days.

6,288,418 B1	9/2001	Reed et al.	
6,415,353 B1 *	7/2002	Leung	711/106
6,937,499 B2 *	8/2005	Nordal et al.	365/145
7,023,844 B2 *	4/2006	Wolrich et al.	370/388
7,209,383 B2 *	4/2007	Dray et al.	365/171

FOREIGN PATENT DOCUMENTS

EP	0 492 862 A2	1/1992
EP	0492862 B1	7/1997
EP	0866632 A2	9/1998
GB	1483903	9/1974
GB	1 483 903	8/1997

(21) Appl. No.: **10/632,885**(22) Filed: **Aug. 1, 2003**(65) **Prior Publication Data**

US 2005/0025196 A1 Feb. 3, 2005

(51) **Int. Cl.**
H04J 3/04 (2006.01)(52) **U.S. Cl.** **370/535**(58) **Field of Classification Search** None
See application file for complete search history.(56) **References Cited**

U.S. PATENT DOCUMENTS

4,596,004 A	6/1986	Kaufman
5,677,638 A	10/1997	Young et al.
5,748,033 A	5/1998	Kaveh et al.
5,905,665 A	5/1999	Rim
5,936,424 A	8/1999	Young et al.

OTHER PUBLICATIONS

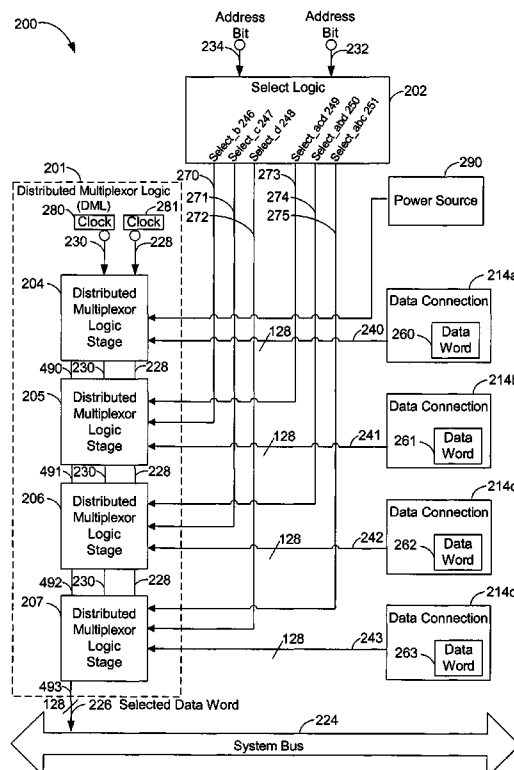
www.ece.msstate.edu/~reese/EE3714/combblocks/combblocks.pdf,
"Combinational Building Blocks", pp. 1-6.

* cited by examiner

Primary Examiner—Andrew C. Lee

(57) **ABSTRACT**

A multiplexor circuit comprising a plurality of data connections, first stage logic configured to receive a first data word from one of the connections and to transmit the first data word received, and second stage logic configured to receive the first data word from the first stage logic and to select a selected data word between the first data word and a second data word received from another of the plurality of data connections based upon a set of select signals, the second stage logic configured to transmit the selected data word.

21 Claims, 7 Drawing Sheets

U.S. Patent

Mar. 25, 2008

Sheet 1 of 7

US 7,349,448 B2

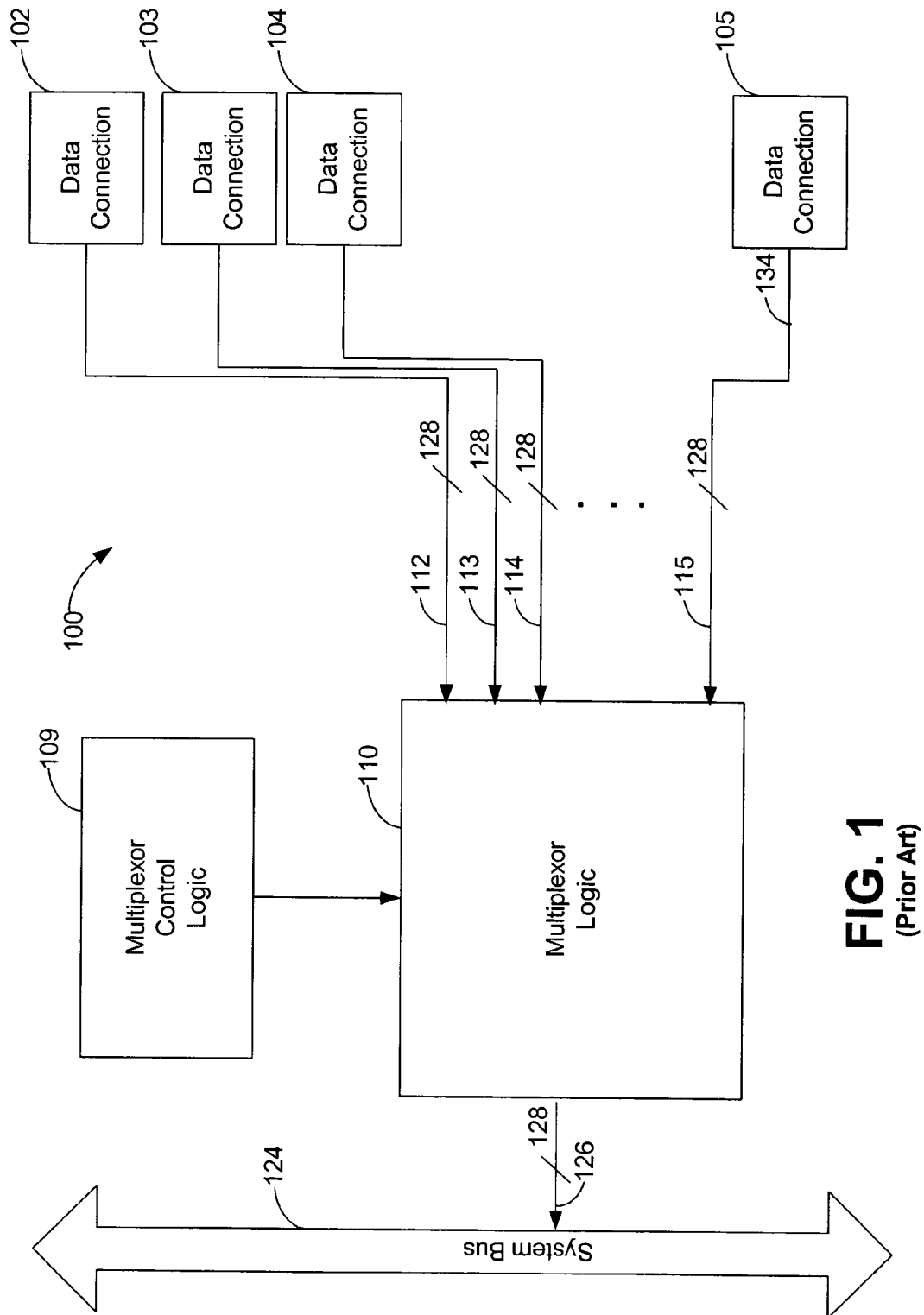


FIG. 1
(Prior Art)

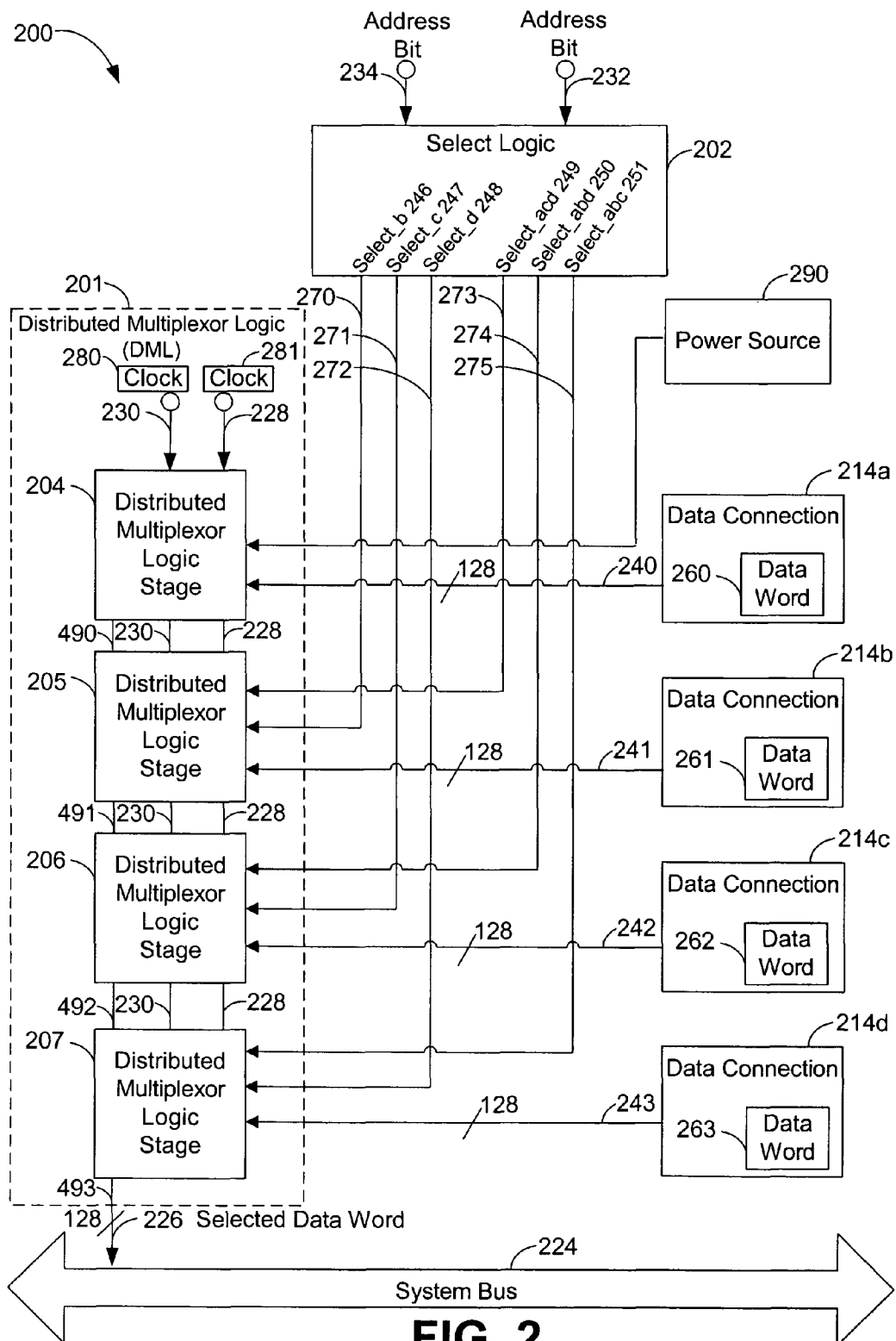


FIG. 2

U.S. Patent

Mar. 25, 2008

Sheet 3 of 7

US 7,349,448 B2

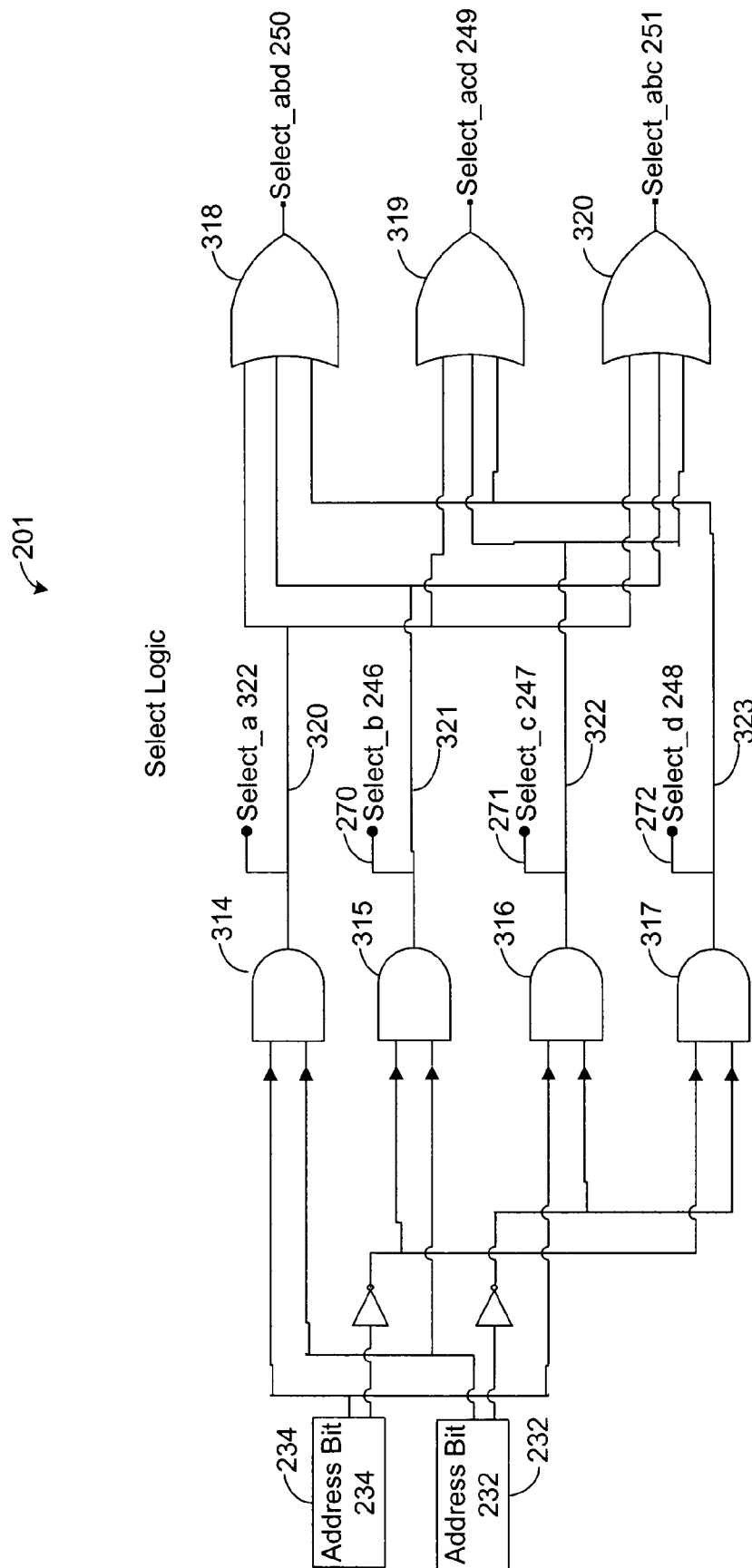
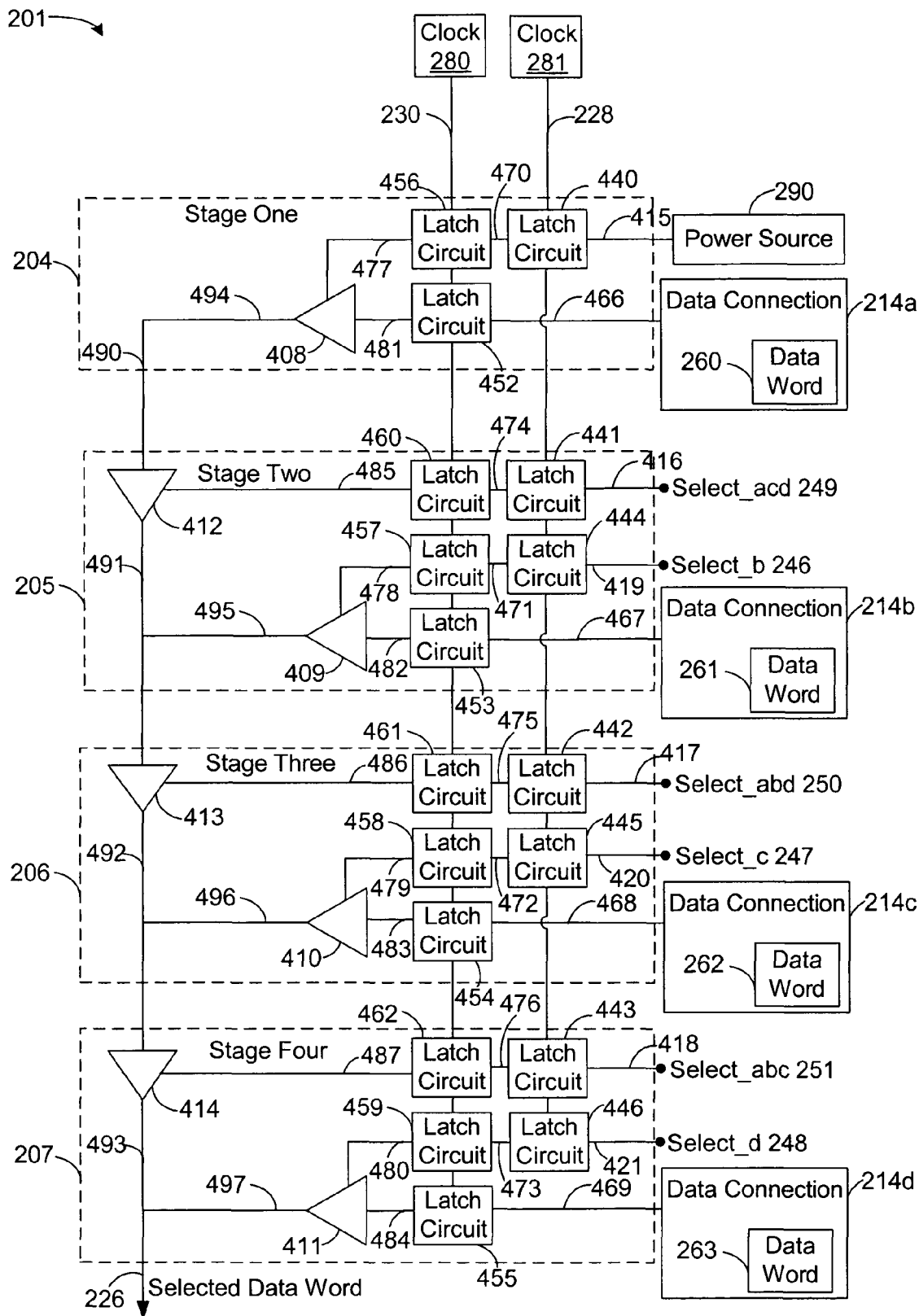


FIG. 3

**FIG. 4**

U.S. Patent

Mar. 25, 2008

Sheet 5 of 7

US 7,349,448 B2

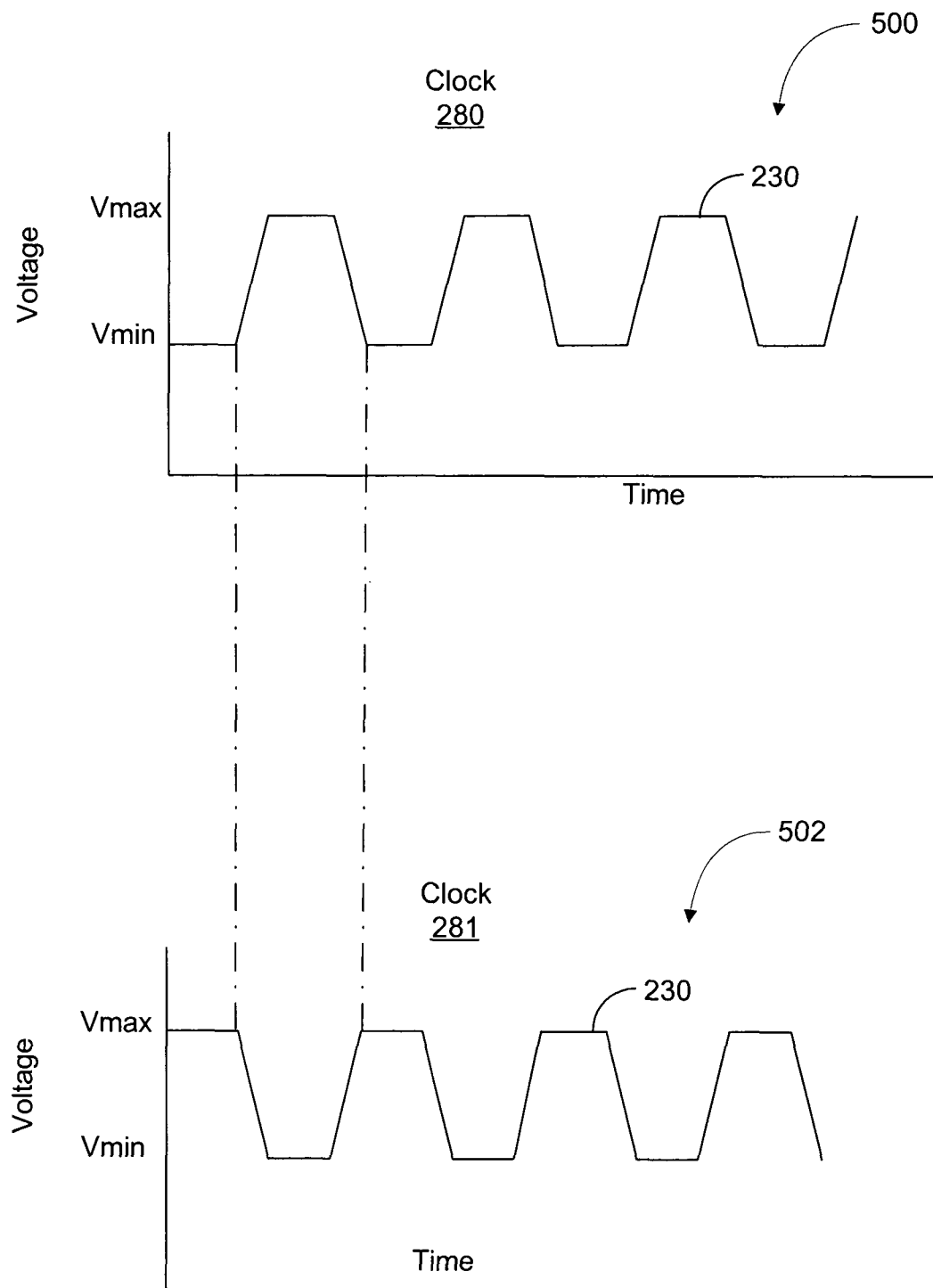


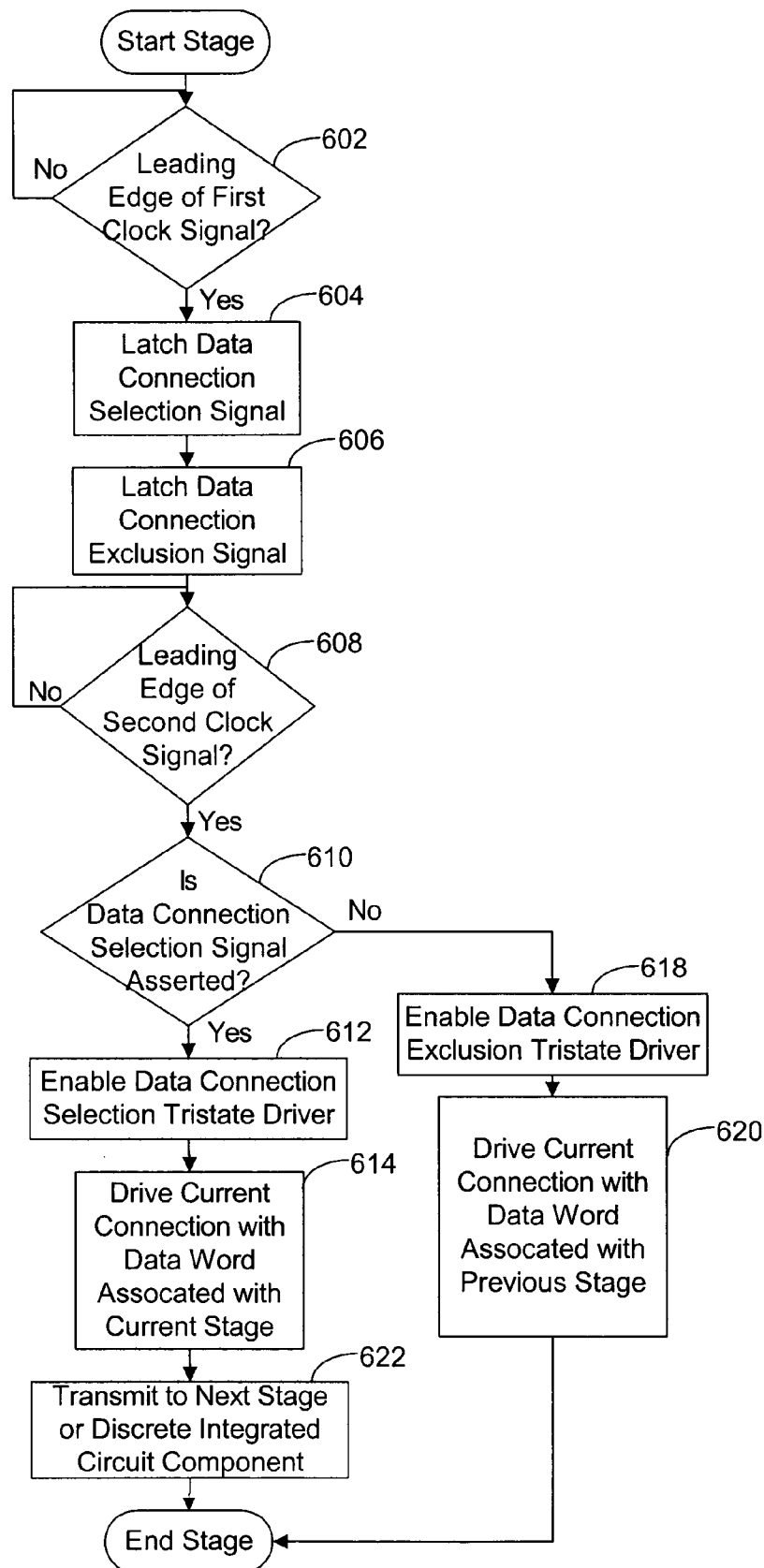
FIG. 5

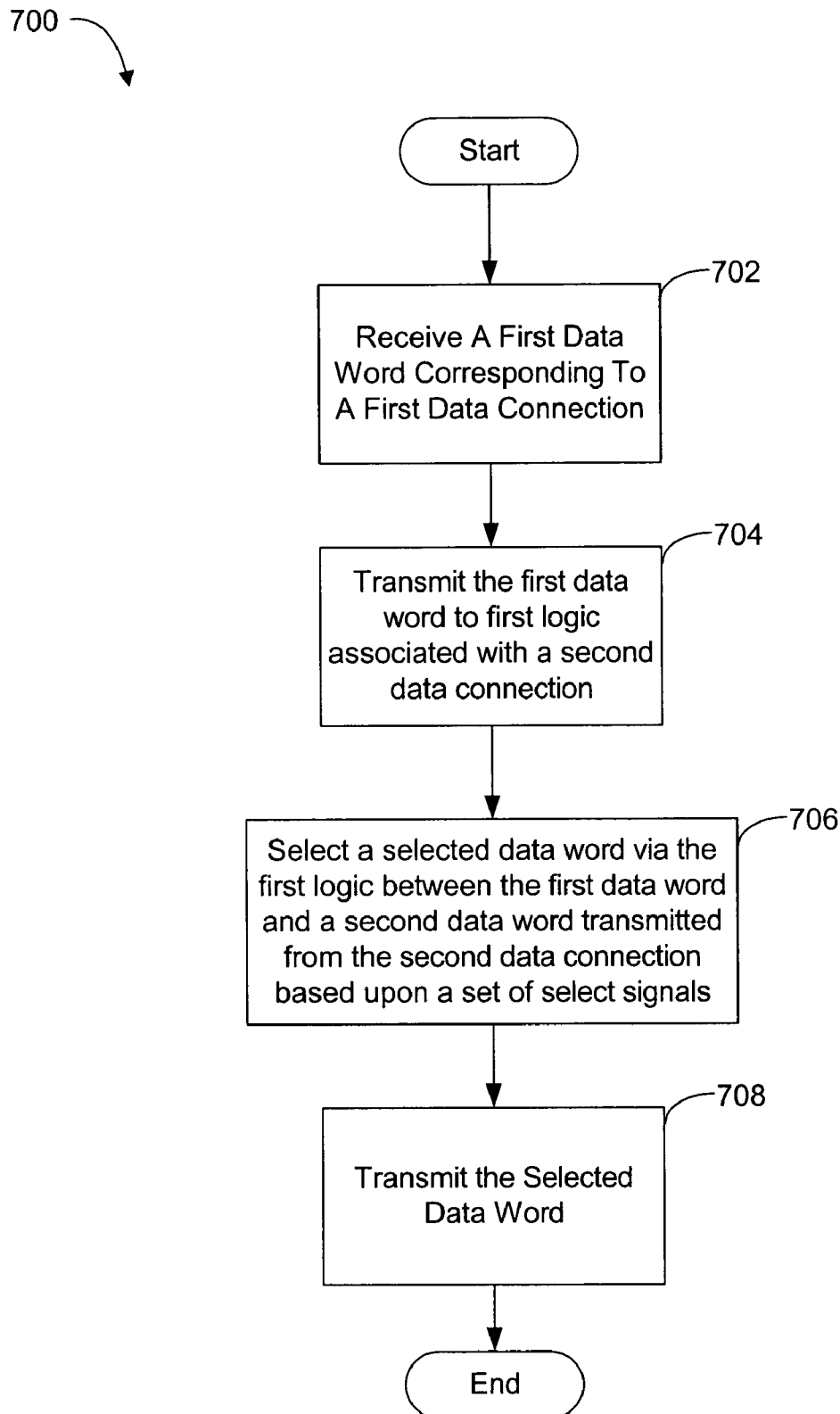
U.S. Patent

Mar. 25, 2008

Sheet 6 of 7

US 7,349,448 B2

**FIG. 6**

**FIG. 7**

US 7,349,448 B2

1

DISTRIBUTED MULTIPLEXING CIRCUIT WITH BUILT-IN REPEATER

BACKGROUND

A standard multiplexor (MUX) circuit is used to select one signal from multiple signals. More specifically, a standard MUX circuit can be implemented on an integrated circuit (IC) and used to select data from a particular one of a plurality of data connections.

For example, a MUX circuit within an IC may comprise eight data connections that transmit 128-bit data words. Typically, each of the 128-bit data words may be routed to multiplexor logic on the IC via a connection that comprises 128 routing wires, e.g., a wire for each bit that is transmitted. Therefore, in this particular example, 1024 routing wires on the IC would be used to route eight data words from eight data connections to the multiplexor logic. The multiplexor logic selects and outputs one of the 128 bit words received by the multiplexor logic. Notably, the aforescribed MUX circuit is prone to routing congestion and the inherent problems associated therewith, e.g., difficulty in designing and fabricating.

A conventional MUX circuit **100** is depicted in FIG. 1. The MUX circuit **100**, of FIG. 1 comprises an array of data connections **102-105** (e.g. data ports, etc.) and multiplexor logic **110**. During operation, each data connection **102-105** transmits a 128-bit data word to multiplexor logic **110** via 128-bit wide data connections **112-115**, respectively.

The multiplexor logic **110** receives the data words and, at any given time, selects one of the data words on one of the connections **112-115** based on a control signal from MUX control logic **109**. The MUX logic **110** transmits the selected data word via an output connection **126**, to an IC component, such as a system bus **124**. The MUX design depicted by FIG. 1 employs a significant number of routing wires, and the number of routing wires increases as the number of data connections increases.

In the past, dynamic MUXs and tristate MUXs have been employed in an attempt to address routing congestion caused by large numbers of routing wires. However, a dynamic MUX typically requires more power, has smaller noise margins, and takes more time to design correctly. Further, usage of tristate multiplexors is sometimes acceptable when the distances between the data connections **102-105** and the multiplexor logic **110** are small. However, speed and noise degradation typically increases as the distances between the data connections increase.

SUMMARY OF THE DISCLOSURE

Embodiments of the present disclosure pertain to systems for distributing multiplexor functionality. One such embodiment is a system that comprises a plurality of data connections and first stage logic configured to receive a first data word from one of the connections and to transmit the first data word received. Further, the system comprises second stage logic configured to receive the first data word from the first stage logic and to select a selected data word between the first data word and a second data word received from another of the plurality of data connections. Further, the second stage logic selects the selected data word based upon a set of select signals and transmits the selected data word.

Another embodiment of the present disclosure comprises a method for distributing multiplexor functionality. One such method comprises the steps of receiving a first data word corresponding to a first data connection, transmitting

2

the first data word to first logic associated with a second data connection, selecting a selected data word via the first logic between the first data word and a second data word transmitted from the second data connection based upon a set of select signals, and transmitting the selected data word.

BRIEF DESCRIPTION OF THE DRAWINGS

The invention can be better understood with reference to the following drawings.

FIG. 1 is a block diagram illustrating a conventional multiplexor.

FIG. 2 is a block diagram illustrating an exemplary embodiment of a distributed multiplexing system of the present disclosure.

FIG. 3 is a block diagram illustrating an exemplary embodiment of select logic, such as is depicted in FIG. 1.

FIG. 4 is a block diagram illustrating an exemplary embodiment of stages 1 through 4 of distributed multiplexor logic, such as is depicted in FIG. 1.

FIG. 5 is a timing diagram that illustrates the relative timing characteristics of an exemplary embodiment of distributed multiplexor logic, such as is depicted in FIG. 1.

FIG. 6 is a flow chart illustrating an exemplary embodiment of architecture and functionality of the distributed multiplexor logic, such as is depicted in FIG. 1.

FIG. 7 is a flow chart illustrating an exemplary embodiment of architecture and functionality of the distributed multiplexor logic, such as is depicted in FIG. 2.

DETAILED DESCRIPTION

Generally, embodiments of a distributed multiplexor circuit select between multiple data connections, each of which is separated from multiplexor logic by some distance on an integrated circuit (IC). Each data connection is preferably capable of transmitting an n-bit data word, where “n” refers to the number of bits within the data word. The n-bit data word is transmitted via a connection that comprises at least n routing wires wherein each different bit of the n-bit word is transmitted via a different routing wire. A set of bits, e.g. 128 bits simultaneously transmitted from the same data connection, is hereinafter referred to as a “data word.” Preferably, each data connection transmits its data word to a distributed multiplexor logic (DML) stage that is associated with the data connection, and select logic then indicates which data word from the multiple data connections is to be output (e.g. transmitted to a discrete IC component).

Each DML stage is one of a plurality of consecutive stages. A first stage in the plurality of consecutive stages preferably receives the data word from its associated data connection. Each of the remaining stages receives a data word from its associated data connection and a data word from a previous stage.

Select logic transmits a set of select signals to each DML stage, and the select signals are indicative of which data word is to be selected for further transmission. In accordance with these select signals, each DML stage selects either the data word it receives from a previous DML stage, if any, or the data word it receives from its associated data connection. In the final stage of the plurality of consecutive stages, the distributed multiplexor circuit drives the resulting selected data word indicated by the select logic onto the system bus or some other type of conductive connection.

FIG. 2 is a block diagram illustrating a distributed multiplexor circuit **200** in accordance with an exemplary embodiment of the present disclosure. The distributed mul-

US 7,349,448 B2

3

tiplexor circuit 200 comprises select logic 202 and DML 201. The circuit 200 further comprises a plurality of data connections 214a-214d, and for brevity and clarity, the multiplexor circuit 200 shall be described hereafter as comprising four data connections, data connections 214a, 214b, 214c, and 214d. However, other numbers of data connections may be employed in other embodiments.

The DML 201 comprises a plurality of stages 204-207, preferably corresponding to the number of data connections, from which a selected data word is chosen and transmitted via connection 226. In a preferred embodiment, the DML 201 comprises a number of stages 204-207 equal to the number of data connections 214a-214d, although other variations in the number of data connections and stages are possible in other embodiments. Thus, the DML 201 of FIG. 2 comprises four stages 204-207, one stage for each data connection 214a-214d. In this regard, logic 201 of circuit 200 comprises DML stages 204, 205, 206, and 207 associated with data connections 214a, 214b, 214c, and 214d, respectively.

During operation of circuit 200, DML stage 204 receives a data word 260 transmitted via a connection 240, from data connection 214a. As described herein, the width of each connection 240-243 corresponds to the width of the data word transmitted by its associated data connection. For example, if data connection 214a transmits a data word that is 128 bits wide, then the connection 240 is preferably comprised of 128 routing wires through which each of the 128 bits of the data word 260 is transmitted. Likewise, DML stage 205 receives a data word 261 transmitted via data connection 241 from data connection 214b, DML stage 206 receives a data word 262 transmitted via connection 242 from data connection 214c, and DML stage 207 receives a data word 263 via connection 243 from data connection 214d.

Moreover, the select logic 202 is communicatively coupled to the DML 201 via routing wires 270-275, and transmits a plurality of select signals 246-251 via these routing wires 270-275 to the DML 201, respectively. The select signals 246-251 indicate which of the data words from data connection 214a-214d, is the “selected data word” that is to be output from the DML 201 and transmitted, via connection 226, to a system bus 224. Note that the select logic 202 may be implemented in a variety of ways, and one such implementation is described in detail hereinafter with reference to FIG. 3.

In the course of operation, the distributed multiplexor circuit 200 depicted in FIG. 2 receives two input bits, address bit 234 and address bit 232. Note that other numbers of address bits may be employed in other embodiments depending upon the number of data connections from which the circuit 200 selects a selected data word for transmission to connection 226. Preferably an application (not shown) requests data from a particular one of the data connections 214a-214d. In so requesting, the application transmits two address bits 232 and 234 having values identifying the data connection 214a-214d from which the application seeks data. For purposes of illustration, the application may assert both address bit 234 and address bit 232, thereby identifying data connection 214b. In response, select logic 202 may assert and/or deassert appropriate select signals to effectuate transmission of data word 261 to the system bus 224 via connection 226.

The select logic 202 asserts select signals 246-251 in accordance with the data connection 214a-214d indicated by the address bits 234 and 232. In particular, the select logic 202 asserts the data connection select signal 246-248 cor-

4

responding to the particular data connection 214a-214d indicated by the address bits 232 and 234. Further, select logic 202 asserts data connection exclusion signals 249-251, to affirm that a particular set of data connections was not selected. In this regard, stage 204-207 associated with the particular data connection 214a-214d uses such exclusion signal 249-251 to drive the data word 260-263 from the previous stage to the next stage, thereby excluding the data word associated with the current stage.

Generally, if circuit 200 is powered via power connection 290, then stage 204 transmits data word 260 to stage 205. Such transmission is clocked by clock signals 230 and 228, which is described in more detail with reference to FIG. 4. Each stage 205-207 thereafter selects from a data word 260-263 transmitted by the previous stage 204-206 and the data word 261-263 received via data connections 240-243. Stage 207 then transmits the final selected data word via connection 226 to the system bus 224. Summarily, each stage 205-207 following stage 204 receives a data word 261-263, a data connection select signal 246-248, and a data connection exclusion signal 249-251, and the signals 246-251 select the data word 261-263 that is to be propagated to the next stage.

For example, if address bits 234 and 232 indicate that the data word associated with data connection 214b is desired, then the select logic 202 asserts the select signal select_b 246, which is received by stage 205 of the DML 201, e.g., the stage associated with data connection 214b. Further, select logic 202 deasserts signals select_c 247 and select_d 248, which are received by stage 206 and stage 207, respectively. Select logic 202 also asserts select_abd 250 and select_abc 251, which are transmitted via connections 274 and 275 to stage 206 and stage 207. Further, select logic 202 deasserts data connection exclusion signals select_acd 249, which is transmitted via connection 273 to stage 205.

In such an example, stage 204 is driven by power source 290, and the data word 260 from the associated data connection 214a is transmitted, by stage 204, to stage 205. Stage 205 receives an asserted data connection selection signal select_b 246 and a deasserted data connection exclusion signal select_acd 249. In response, DML stage 205 disables throughput of data word 260 from stage 204 to stage 206, and, alternatively transmits data word 261 from the data connection 214b to stage 206. The signals received by stage 206 include data word 262 associated with data connection 214c, a deasserted data connection select signal select_c 247, and an asserted data connection exclusion signal select_abd 250. Thus, stage 206 enables transmission of the data word 261 associated with data connection 214b to be transmitted to stage 207. Likewise, in accordance with select signals received by select logic 202, stage 207 also enables transmission to the system bus data word 261 associated with data connection 214b as the selected data word via connection 226. Thus, DML 201 transmits, to the system bus 224, the data word associated with the data connection 214a-214d indicated by the address bits 234 and 232.

Note that the DML 201 further comprises clocks 280 and 281, which transmit clock signals 230 and 228. These clock signals 230 and 228 control the timing of the data connection selection effectuated by the DML 201. The DML 201 receives the data words 260-263 from the data connections 214a-214d and the select signals 246-251 from the select logic 202 and performs selection of the data words 260-263 in accordance with the cycles of the clock signals 230 and 228, as will be described in more detail hereinbelow.

As previously discussed herein, the select logic 202 may receive from application logic (not shown) address bit 234

US 7,349,448 B2

5

and address bit 232. Address bits 234 and 232 exhibit values indicative of a particular data connection, i.e., 214a, 214b, 214c, and 214d. The following truth table illustrates exemplary address bit values and the data connection indicated by the particular address bit pairs. Note that a bit pair refers to the simultaneously received dual values of address bit 234 and address bit 232.

TABLE 1

Address Bit 234	Address Bit 232	Data connection			
		214a	214b	214c	214d
1	1	1	0	0	0
1	0	0	1	0	0
0	1	0	0	1	0
0	0	0	0	0	1

With reference to Table 1, if the application (not shown) seeks data from data connection 214a, then, in accordance with the truth table, it asserts address bit 234 and address bit 232. If it seeks data from data connection 214b then it asserts address bit 234 and deasserts address bit 232. If it seeks data from data connection 214c then it deasserts address bit 234 and asserts address bit 232, and if it seeks data from data connection 214d it deasserts both address bits 234 and 232.

An exemplary, implementation of the select logic 202, is now described with reference to FIG. 3. In accordance with the following truth table the select logic 202 can be implemented with complementary metal semiconductor (CMOS):

TABLE 2

CONNECTION DESIRED	select d 248	select abc 251	select b 246	select acd 249	select c 247	select abd 250
214a	0	1	0	1	0	1
214b	0	1	1	0	0	1
214c	0	1	0	1	1	0
214d	1	0	0	1	0	1

Note that the select logic 202 transmits two types of “select signals,” data connection selection signals 246-248 and data connection exclusion signals 249-251. A data connection selection signal 246-248 is asserted in accordance with the data connection 214a-214d indicated by the address bits 234 and 232. Thus, if the address bits 234 and 232 indicate selection of data connection 214d then select_d 248 (FIG. 2) is asserted. Further if address bits 234 and 232 indicate data connection 214c, then select_c 247 is asserted, and if they indicate data connection 214b, then select_b 246 is asserted.

Note that select signal select_a is not addressed in Table 2, and in the exemplary embodiment described hereafter, the select signal select_a is not needed. Other embodiments of select logic 202, however, may use such a select signal select_a. The reason for the omission of the select signal select_a from Table 2 is that as described herein, DML stage 204 does not require a select signal from select logic 202. This lack of requisition of a select signal is because stage 204 does not have a preceding stage from which it receives a data word 260-263. Therefore, as will become apparent by the following description, stage 204 does not need an indication from select logic 202 of whether to choose the data word from the preceding stage or the data word from the data connection 214a associated with stage 204.

Table 2 indicates assertion and deassertion by the select logic 202 of the select signals in order to select a particular

6

data connection. For example, if data connection 214b is the desired connection, the data connection select signal select_b 246 (FIG. 2) is asserted, and the select signals select_d 248 and select_c 247 are deasserted. In addition, data connection exclusion signals select_abc 251 and select_abd 250 are asserted, and select_acd 249 is deasserted.

In addition to data connection select signals 246-248, select logic 202 also asserts and/or deasserts data connection exclusion signals 249-251. Exclusion signals 249-251 are also asserted and/or deasserted in accordance with the address bits received. However, the exclusion signals 249-251 identify which data connections are not being selected, i.e., which data connections are being excluded. For example, if the address bits 234 and 232 indicate that data connection 214d is sought, then in addition to asserting the select signal select_d 248, select_abc 251 is deasserted and transmitted to stage 207, which is associated with data connection 214d, and indicates that data connections 214a, 214b, 214c are excluded. Likewise, select_abd 250 and select_acd 249 are asserted indicating that data connection 214b is excluded and data connection 214c is excluded and transmitted to stage 205 and stage 206, respectively.

FIG. 3 illustrates an exemplary complementary metal oxide semiconductor (CMOS) implementation of the truth table of Table 2. Note that other implementations of the truth table in other various forms of hardware are possible in other embodiments. However, the embodiment of the select logic 202 in FIG. 3 comprises two inputs, i.e., two address bits 234 and 232. The select logic 201 further comprises a plurality of logical gates 314-317. The number of logical gates 314-317 preferably corresponds to the number of data connections 214a-214d (FIG. 2) in the distributed multiplexor circuit 200 (FIG. 2).

The exemplary implementation of select logic 202 of FIG. 3 comprises four exemplary logical AND gates 314-317, which transmits four data connection select signals select_a 322, select_b 246, select_c 247, and select_d 248 via connections 320-323 indicating which data connection is sought by an application. Further, select_b 246, select_c, 247, and select_d 248 are transmitted to DML 201 (FIG. 2) via connections 270-272.

In accordance with Table 1, the select logic 202 asserts select_a 322 when address bit 234 is asserted and address bit 232 is asserted, asserts select_b 246 when address bit 234 is asserted and address bit 232 is deasserted, asserts select_c 247 when address bit 234 is deasserted and address bit 232 is asserted, and asserts select_d 248 when address bit 234 is asserted and address bit 232 is asserted.

The select logic 202 transmits the data connection selection signals 246-251 via wires 270-272 to their corresponding stages 204-207. In this regard, select_b 246 is transmitted to stage 205, which is associated with data connection 214b. In addition, select_c 247 is transmitted to stage 206, which is associated with data connection 214c, and select_d 248 is transmitted to stage 207, which is associated with data connection 214d.

Moreover, select logic 202 uses the select signals transmitted from the plurality of AND gates 314-317 via connections 320-323 as input to a plurality of OR gates 318-320 for ascertaining the data connection exclusion signals 250-251 for the distributed multiplexor circuit 200 (FIG. 2). In the embodiment depicted in FIG. 3, three OR gates 318-320 are employed. Select logic 202 asserts data connection exclusion signal select_abd 250 when the select_c 247 is asserted via AND gate 318. In addition, the select logic 202 asserts the data connection exclusion signal select_acd 249 when the data connection select signal select_b 246 is

US 7,349,448 B2

7

asserted via AND gate 315. Finally, the select logic 202 asserts the data connection exclusion signal select_abc 251 when the data connection select signal select_d 248 is asserted via AND gate 317. Select logic 202 then transmits the data connection exclusion signals 249-251 to its corresponding DML stage (FIG. 2) 204-207.

As described above, select_abd 250, when asserted, indicates that data connections 214a, 214b and 214d are excluded (i.e., data connection 214c is requested), and select_acd 249, when asserted, indicates that data connection 214a, 214c and 214d are excluded (i.e., data connection 214b is requested). Further select_abc 251, when asserted, indicates that data connections 214a, 214b, and 214c are excluded, (i.e., data connection 214d is requested).

FIG. 4 illustrates an exemplary implementation of the DML 201 of FIG. 2. Each DML stage 204-207, preferably comprises data connection selection tristate drivers 408-411, each of which receives and outputs its associated data word 260-263 to its associated connection 490-493, respectively, based upon select signals 246-251.

In addition to receiving its associated data word 260-263, each data connection selection tristate driver 408-411 also receives an enable signal, which is an output 477-480 from a latch circuit 456-459. Such output 477-480 drives the data connection selection tristate driver 408-411. In this regard, if the output 477-480 is asserted via the corresponding latch circuit 456-459 then the output 477-480 enables the data connection selection tristate driver 408-411 to which the output 477-480 is connected. If the data connection selection tristate driver 408-411 is enabled, then the data word 260-263 associated with the data connection selection tristate driver 408-411 drives its associated connection 490-493 via connection 494-497.

Each of stages 205-207 further comprises a data connection exclusion tristate driver 412-414, which is configured to receive the data word 260-262 that is driven to the previous stage's bus 490-492. Additionally, each data connection exclusion tristate driver 412-414 receives an enable signal, which is an output 485-487 of a latch circuit 460-462. Such output 485-487 drives the data connection exclusion tristate driver 412-414. If the output 485-487 is asserted via its corresponding latch circuit 460-462, then the output 485-487 enables the data connection exclusion tristate driver 412-414 to which the output 485-487 is connected. If the data connection exclusion tristate driver 412-414 is enabled, then the data word 260-263 currently driving the previous connection 490-492 drives the current connection 491-493.

Thus, each of the stages 205-207 effectively selects which signal drives its connection 491-493. If data connection selection tristate driver 408-411 is enabled, then the data word 260-263 of the data connection 214a-214d drives its associated connection 490-493. If the current stage's data connection exclusion tristate driver 412-414 is enabled, and thus the current stage's data connection selection tristate driver 408-411 is disabled, then the stage 205-207 selects the data word 260-263 currently driving the previous stage 204-206 to drive the current stage 205-207, respectively. Furthermore, whether data connection selection tristate drivers 408-411 are enabled or data connection exclusion tristate drivers 412-414 are enabled depends upon select signals 246-251.

The timing of the selection by the DML 201 of which data word 260-263 is transmitted to connections 490-493 and ultimately to the system bus 224 via connection 226 is controlled via clock signals 230 and 228 transmitted by clock 280 and clock 281.

8

Hence, the output 470-476 of each latch circuit 440-446 changes upon the detection of a leading edge of clock signal 228, and the value of the output 470-476 latched by the latch circuit 440-446 reflects the current value at the input 415-421 of the latch circuit 440-446 upon detection of the leading edge of the assertion of the clock signal 228. If during such assertion of the clock signal 228 the value at the input 415-421 changes, then such a change is reflected in the output 470-476. When the clock signal 228 is deasserted, the value currently at the input 415-421 is latched at the output 470-476 until another leading edge of clock signal 228 activates the latch 440-446.

Likewise, each output 477-487 of each latch circuit 452-462 changes upon the detection of a leading edge of clock signal 230, and the value of the input 466-476 is latched at the output 477-487 by the latch circuit 452-462 upon detection of the leading edge of the assertion of the clock signal 230. If during such assertion of the clock signal 230, the value at the input 466-476 changes, then such a change is reflected in the output 477-487. When the clock signal 230 is deasserted, the value currently at the output 477-487 is latched until the another leading edge of clock signal 230.

Therefore, during operation, each data connection 214a-214d preferably transmits a data word 260-263, which is present as input 466-469 at a corresponding latch circuit 452-455. Further, select logic 202 (FIG. 2) transmits to each of stages 205-207 a set of select signals including a data connection select signal 246-248 and a data connection exclusion signal 249-251. Each data connection select signal 246-248 is present as input 419-421 at the corresponding latch circuit 444-446, and each data connection exclusion signal 249-251 is present as input 416-418 at the corresponding latch circuit 441-443.

When clock signal 228 is asserted, the leading edge of the clock signal 228 triggers latch circuits 440-446. When the latch circuits 440-446 are triggered, the data presently at the inputs 415-421 are latched as the outputs 470-476. Thus, in stage 204, if power source 290 is providing a power signal, such value indicative of the power source 290 is latched by latch circuit 440 as output 470.

Likewise, as will be described further herein with reference to FIG. 5, clock signals 228 and 230 preferably operate out-of-phase, such that when clock signal 228 is deasserted, then clock signal 230 is asserted, thereby triggering latch circuits 452-462. Therefore, the outputs 470-476 are latched by latch circuits 440-446 and remain latched to serve as inputs to latch circuits 452-462. When the leading edge of clock signal 230 triggers latch circuits 452-462, then those outputs 470-476 simultaneously are latched as the outputs 477-487. Thus, data connection selection tristate driver 408 is enabled by output 477, if power is being supplied to the DML 201, which results in data word 260 driving connection 490. Hence, data word 260 is present as input to data connection exclusion tristate driver 412.

Simultaneously therewith, the values of the select signals 246-251 are latched for each stage 205-207 and data words 260-264 are latched for each stage 204-207. Therefore, in accordance with the select signals 246-251 the data word 260-263 indicated by the select signals is transmitted as the selected data word via connection 226.

Specifically, if select_acd 249 is asserted, then the output 485 of latch circuit 460 when the clock signal 230 triggers latching the value of the asserted data connection exclusion signal select_acd 249 is asserted, thus enabling data connection exclusion tristate driver 412. If data connection exclusion tristate driver 412 is enabled, then data word 260 drives connection 491 of stage 205 via data connection

US 7,349,448 B2

9

exclusion tristate driver **412**. The data connection exclusion tristate driver **412** further serves as a repeater when driving such data word **260** to the connection **491**, and, therefore, increases the signal strength of the data word **260**.

Alternatively, if select_acd **249** is deasserted, then the output **485** of latch circuit **460** is deasserted. Furthermore, in accordance with Table 2 hereinabove, if select_acd **249** is deasserted, then select_b is asserted, indicating that the data word **261** is desired. Therefore, the output **478** of latch circuit **457** is asserted, thus enabling data connection selection tristate driver **409**. When data connection selection tristate driver **409** is enabled, then data word **261** from data connection **214b** drives connection **491** via output **495**.

Each stage **206-207** thereafter behaves in the manner described with reference to stage **205**. For brevity, each stage will not be described further in detail. Note, however, that as each stage **205-207** performs selection between the data word associated with the stage and the data word driving the previous connection, that the final stage connection **493** is driven by the data word **260-263** selected by the select signals **246-251** received from select logic **201**.

As indicated in FIG. 4, stage **204** does not receive a data word from a previous stage, nor does it receive select signals, for reasons discussed herein. Therefore, the data word **260** is driven to the connection **490** when power is supplied to DML **201**. In this regard, the tristate driver **408** is enabled via latch circuits **456** and **440** from power source **290** via clock signals **230** and **228** from clocks **280** and **281**, respectively.

Clocks **280** and **281** are preferably configured to operate at opposite phases. Consequently, when signal **230** of clock **280** is asserted, signal **228** of clock **281** is deasserted, and when signal **228** of clock **281** is asserted, signal **230** of clock **280** is deasserted. As illustrated in FIG. 5, diagram **500** illustrates a signal **230** for clock **280** operating at maximum voltage, e.g., being asserted. As indicated in diagram **502**, when signal **230** is being asserted, signal **228** of clock **281** is deasserted, e.g., operating at minimum voltage. As shown, at each cycle clock **280** and **281** cycle to the opposing phase.

Therefore, when clock signal **228** is asserted, latch circuits **440-446** connected to clock **281** and controlled via clock signal **228** allow signals to propagate from the input of the latch circuits **440-446** to the outputs **470-476** of latch circuits **440-446**. If inputs **415-421** of latch circuits **440-446** change any time that clock signal **228** of clock **281** is asserted, then it will be reflected in the output. When clock **281** is deasserted, the most recent value that is latched by latch circuits **440-446** will be held until the next cycle, e.g., until clock signal **228** is asserted again. Likewise, latch circuits **452-462** behave in the same manner in accordance with clock signal **230** from clock **280**. In this regard, at the leading edge of clock signal **230** outputs **470-476** present at the input of latch circuits **452-462** are latched as outputs **477-487** of latch circuits **452-462**.

The final stage **207** transmits a selected data word, which is one of the data words **260-263** to the system bus **224** (FIG. 2). Thus, the selected data word **260-263** is directly driven through the connections **490-493** of the DML **201** and transmitted as selected data word **226**, which is selected via select signals **246-251** by select logic **202**.

In propagating the data words **260-263** from the first stage to the final stage of the circuit **201**, the tristate drivers **412-414** located between each stage **204-207** serve to increase the signal strength of the data word propagated from the previous stage, thereby acting as repeaters. These

10

tristate drivers **412-414** therefore allow the data connections **214a-214d** to be separated by a relatively long distance on an IC.

Note that it is preferable that the select signals **246-251**, which are transmitted from select logic **202** for controlling the tristate drivers **408-411** and **412-414** all enable and disable the tristate drivers concurrently. The circuit **201** (FIG. 2) accomplishes concurrent enablement and disablement of the latch circuits **440-446** and latch circuits **452-462** and the timing thereof by clocks **280** and **281**. Concurrent enablement and disablement of the tristate drivers **408-411** and **412-414** limit the amount of "drive fight," e.g., the time that more than one driver is attempting to drive a connection **490-493** to a different data word **260-263**.

Operation of an exemplary architecture and functionality of a stage **205-207** of DML **201** is now discussed with reference to FIG. 6. As described and explained hereinabove, stage **204** preferably behaves somewhat differently than the remaining stages **205-207**. Note that other embodiments of DML **201** may employ greater or fewer than the four stages **204-207** illustrated in the embodiment of DML **201** in FIG. 4.

An exemplary stage **205-207** preferably operates in accordance with clock signals **230** and **228** originating with clocks **280** and **281**, respectively. The aforementioned clock signals **230** and **228** preferably operate out of phase, e.g., when one such clock is asserted, the other one is deasserted.

A stage **205-207** latches a data connection selection signal, as indicated in step **604**, and a data connection exclusion signal **606**, as indicated in step **606**, when a first clock signal is asserted, e.g., when a clock signal transitions from a minimum voltage value to a maximum voltage value, as indicated in step **602**. Note that latching occurs, as described hereinabove, when a digital logic circuit that comprises a clock input, a data input, and a data output, is used to store one or more bits. Generally, when the clock input is asserted, then data on the input is exhibited and retained on the circuit output, e.g., the data is latched on the circuit's output. When the clock goes inactive, the data is retained on the circuit output until the clock goes active again.

When a second clock signal is asserted, as indicated in step **608**, the latched values are used to effectuate selection of a data word. In this regard, if the data connection selection signal latched in step **604** is asserted, as indicated in step **610**, then a data connection selection tristate driver **409-411** (FIG. 4) is enabled, as indicated in step **612**.

Thus, the DML **201** (FIG. 4) transmits the data word associated with the current stage to the connection **491-493** (FIG. 4), as indicated in step **614**, via the data connection selection tristate driver **409-411**. By virtue of connecting the current connection **490-493** to either another connection **490-493** of another stage or by connecting the current connection **490-493** to a system bus, the next step is transmitting the selected data word **260-263** from the current stage. Transmission from the current stage can be to a next stage or to a system bus, as described herein and as indicated in step **622**.

If the data connection selection signal is not asserted, then the data connection exclusion signal is asserted, which enables the data connection exclusion tristate driver **412-414**, as indicated in step **618**. Thus, the data word from the previous stage is transmitted to the current stage's connection **491-493** via the data connection exclusion tristate driver **412-414**, as indicated in step **620**, and the data word cur-

US 7,349,448 B2

11

rently being driven to the connection 491-493 is transmitted either to a next stage or to the system bus, as indicated in step 622.

To further illustrate DML 201, if an application requests data word 262 from data connection 214c, select logic 201 (FIG. 2) asserts select_c 247 (FIG. 4), select_acd 249 (FIG. 4), and select_abc 251 (FIG. 4). Further in accordance with the application's request, select logic 201 deasserts select_b 246 (FIG. 4), select_abd 250 (FIG. 4), and select_d 248 (FIG. 4). Stage 204 (FIG. 4) receives a power signal and thus drives connection 490 (FIG. 4) with data word 260 (FIG. 4) from data connection 214a (FIG. 4).

Note that each stage 205-207, thereafter, by enablement or disablement of its associated data connection selection tristate drivers 409-411 and data connection exclusion tristate drivers 412-414 selects whether to drive its connection 491-494 with its associated data word 261-263 or the data word from the previous stage. Enablement and/or disablement of select tristate drivers 408-414 is effectuated by select signals 246-251 and in accordance with clock signals 228 and 230.

At the leading edge of clock signal 228 (FIG. 4), data connection selection signals select_b 246, select_c 247, and select_d 248 are latched, as indicated by decision step 602 and step 604. Select_b 246 is latched as output 471 (FIG. 4), select_c 247 is latched as output 472 (FIG. 4), and select_d is latched as output 473 (FIG. 4). Further, data connection exclusion signals select_acd 249, select_abd 250, and select_abc 251 are latched, as indicated by step 606.

When the first clock 281 cycles and deasserts its clock signal 228 and upon the leading edge of a second clock signal 230, the data connection exclusion tristate driver 412 is enabled in stage 205 by the output 485 from latch circuit 460, because the data connection exclusion signal select_acd 249 is asserted, as indicated by decision step 610 and step 618. Thus, the data word 260 drives the stage 205 connection 291, as indicated in step 620. Accordingly, select_b 246 disables tristate driver 409, thus allowing the connection 491 to be driven by the data word 260 from the previous stage 204.

Preferably contemporaneously, upon the leading edge of the same second clock signal 230, as indicated in step 608, and at stage 206, the data connection selection tristate driver 410 is enabled by the output 479 from latch circuit 458, because data connection selection signal select_c 247 is asserted, as indicated by decision step 610 and step 612. Thus, the data word 262 drives the stage 206 connection 292, as indicated in step 614. Accordingly, select_abd 250 is deasserted, thereby allowing the connection 492 to be driven by the data word 262 from the current stage 206.

Also preferably contemporaneously therewith, upon the leading edge of the same second clock signal 230, as indicated in step 608, and at stage 207, the data connection exclusion driver 414 is enabled by the output 487 from latch circuit 462, because the data connection exclusion signal select_abc 251 is asserted, as indicated in decision step 610 and step 618. Thus, the data word 262 drives the stage 207 connection 293, as indicated in step 620. Accordingly, select_d 248 disables tristate driver 411, thus allowing the connection 493 to be driven by the data word 262 from the previous stage 206, as indicated in step 622.

The finally selected data word, which in the example provided is data word 262, is then transmitted to a system bus 224 via a connection 226.

Note that similar designs can be made with any number of stages. Further note that if the distance and the number of stages increases beyond what can be done in a single timing

12

iteration by clocks 280 and 281, the design can be spread across multiple timing iterations using a additional electronic components, e.g. latches, flip flops and/or other devices.

FIG. 7 describes generally a method 700, which illustrates an exemplary architecture and functionality of DML 201 (FIG. 2). As indicated in step 702, a DML stage 204 receives a first data word 260 corresponding to a first data connection 214a. DML stage 204 transmits the first data word 260 to first logic DML stage 205 associated with a second data connection 214b, as indicated in step 704.

DML stage 205 then proceeds to select a selected data word between the first data word 260 and a second data word 261 transmitted from the second data connection 214b based upon a set of select signals, as indicated in 706. As described herein DML stage 205 selects which data word to throughput to DML stage 206 in accordance with select signals select_acd 249 and select_b 246. DML logic stage 205 then transmits the selected data word, either data word 260 or data word 261, as indicated in step 708.

Now, therefore, the following is claimed:

1. A circuit for multiplexing a plurality of data words including a first data word and a second data word, comprising:

a plurality of data connections;

first stage logic having a first tristate driver, the first tristate driver configured to receive the first data word from one of the connections and to transmit the first data word received; and

second stage logic configured to receive the first data word from the first stage logic and to receive the second data word from another of the plurality of data connections, the second stage logic configured to select for transmission one of the first data word and the second data word based upon a set of select signals, the second stage logic configured to transmit the data word selected for transmission by the second stage logic, the second stage logic having a connection, a second tristate driver, and a third tristate driver, the second tristate driver having an output electrically coupled to an output of the third tristate driver, the second tristate driver configured to receive the first data word from the first tristate driver and to drive the connection of the second stage logic with the first data word if the first data word is selected for transmission by the second stage logic, the third tristate driver configured to receive the second data word and to drive the connection of the second stage logic with the second data word if the second data word is selected for transmission by the second stage logic.

2. The circuit of claim 1, wherein the second stage logic is further configured to transmit, to a third stage logic, the data word selected for transmission by the second stage logic.

3. The circuit of claim 1, wherein the second stage logic is further configured to transmit, to a system bus, the data word selected for transmission by the second stage logic.

4. A multiplexor circuit, comprising:

a plurality of data connections;

first stage logic configured to receive a first data word from one of the connections and to transmit the first data word received based on a set of select signals; and second stage logic configured to receive the first data word from the first stage logic and to select a selected data word between the first data word and a second data word received from another of the plurality of data connections based upon a set of select signals, the

US 7,349,448 B2

13

second stage logic configured to transmit the selected data word to a third stage logic and to increase a signal strength of the first data word when the first data word is transmitted, if the first data word is selected.

5. A multiplexor circuit, comprising:

a plurality of data connections;

first stage logic configured to receive a first data word from one of the connections and to transmit the first data word received; and

second stage logic configured to receive the first data word from the first stage logic and to select a selected data word between the first data word and a second data word received from another of the plurality of data connections based upon a set of select signals, the second stage logic configured to transmit the selected data word to a third stage logic,

wherein the second stage logic comprises:

a first tristate driver configured to drive a connection with the first data word from the first stage logic, the first tristate driver enabled if a data connection exclusion signal is asserted and disabled if the data connection exclusion signal is deasserted; and

a second tristate driver configured to drive the connection with the second data word, the second tristate driver enabled if a data connection select signal is asserted and disabled if the data connection select signal is deasserted.

6. The circuit of claim 5 wherein the first tristate driver increases the signal strength of the first data word when the selected data word is the first data word.

7. The circuit of claim 5, further comprising:

a first latch circuit configured to receive the data connection exclusion signal, the first latch circuit configured to latch the value indicative of the data connection exclusion signal at an output of the first latch circuit, the output of the first latch circuit configured to enable the first tristate driver if the data connection exclusion signal is asserted;

a second latch circuit configured to receive the data connection selection signal, the second latch circuit configured to latch the value indicative of the data connection selection signal at an output of the second latch circuit, the output of the second latch circuit configured to enable the second tristate driver if the data connection selection signal is asserted;

a third latch circuit configured to receive the second data word, the third latch circuit configured to latch the second data word at an output of the third latch circuit, the output of the third latch circuit configured for receipt by the second tristate driver;

a first clock configured to transmit a clock signal to the first, second, and third latch circuits, the clock signal activating latching of the second data word, the data connection selection signal and the data connection exclusion signal at the outputs of the third, second and first latch circuits, respectively.

8. A circuit comprising

a plurality of data connections;

logic associated with a first data connection of the plurality of data connections, the logic associated with the first data connection configured to transmit a first data word corresponding to the first data connection;

logic associated with a second data connection of the plurality of data connections configured to receive the first data word corresponding to the first data connection from the logic associated with the first of the plurality of data connections, the logic associated with

14

the second data connection configured to select a first selected data word between the first data word and a second data word corresponding to the second data connection based upon a set of select signals; and

logic associated with a third data connection of the plurality of data connections configured to receive the first selected signal set, the logic associated with the third data connection configured to select a second selected data word between the first data word and a third data word corresponding to the third data connection based upon a set of select signals, the logic associated with the third data connection further configured transmit the second selected data word to a system bus.

9. A multiplexor circuit, comprising:

a first tristate driver comprising a first data input, a first data output, and a first enable input, the first tristate driver adapted to receive a first data word via the first data input representative of a first data connection;

a second tristate driver comprising a second data input, a second data output, and a second enable input, the second tristate driver adapted to receive a second data word representative of a second data connection;

a third tristate driver having an output coupled to the first data input of the first tristate driver; and

a connection adapted to receive, via the data output of the first tristate driver, the first data word, if a first signal is asserted at the first enable input of the first tristate driver, the connection further adapted to receive, via the second data output of the second tristate driver, the second data word, if a second signal is asserted at the enable input of the second tristate driver.

10. The multiplexor circuit of claim 9, further comprising select logic configured to assert the first signal if an application program desires the first data word representative of the first data connection and configured to assert the second signal if the application program desires the second data word representative of the second data connection.

11. The multiplexor circuit of claim 10, wherein the select logic deasserts the first signal if the application program does not desire the first data word and deasserts the second signal if the application program does not desire the second data word.

12. The multiplexor circuit of claim 11, wherein the first signal and the second signal enable and disable the first and second tristate drivers simultaneously.

13. The multiplexor circuit of claim 12, further comprising:

a first latch circuit configured to transmit the first signal to the first tristate driver upon a leading edge of a clock signal; and

a second latch circuit configured to transmit the second signal to the second tristate driver upon the leading edge of the clock signal.

14. A method for distributing multiplexing, comprising: transmitting a first data word from a first tristate driver to a second tristate driver;

receiving a second data word; and

selecting for transmission between the first data word and the second data word based upon a set of select signals; driving a first connection with the first data word via the second tristate driver if the first data word is selected by the selecting; and

driving the first connection with the second data word via a third tristate driver if the second data word is selected by the selecting.

US 7,349,448 B2

15

15. The method of claim **14**, wherein the selected data word is transmitted to a discrete integrated circuit (IC) component.

16. The method of claim **15**, wherein the select signals comprise a data connection selection signal indicative of which data word is to be selected and a data connection exclusion signal indicative of which data word is to be excluded.

17. A method for distributing multiplexing, comprising:
 receiving a first data word corresponding to a first data connection;
 transmitting, via a tristate driver, the first data word to first logic associated with a second data connection;
 selecting a selected data word via the first logic between the first data word and a second data word transmitted from the second data connection based upon a set of select signals; and
 transmitting the selected data word,
 wherein the selecting comprises:
 driving the second data word to a connection via a first tristate driver if the data connection selection signal is asserted; and
 disabling a second tristate driver configured to receive the first data word and configured to drive the first data word to the connection if the data connection exclusion signal is asserted.

18. The method of claim **17**, wherein the selecting step further comprises:
 driving the first data word to the connection via the second tristate driver if the data connection exclusion signal is asserted; and
 disabling the first tristate driver if the data connection selection signal is deasserted.

19. The method of claim **18**, wherein the driving the first data word to the connection further comprises increasing signal strength of the first data word.

16

20. The circuit of claim **1**, wherein the plurality of data words includes a third data word, the circuit further comprising third stage logic configured to receive the third data word and the data word selected for transmission by the second stage logic, the third stage logic configured to select for transmission one of the third data word and the data word selected for transmission by the second stage logic based on a set of select signals, the third stage logic having a connection, a fourth tristate driver, and a fifth tristate driver, the fourth tristate driver having an output electrically coupled to an output of the fifth tristate driver, the fourth tristate driver configured to receive the data word selected for transmission by the second stage logic and to drive the connection of the third stage logic with the data word selected for transmission by the second stage logic if the data word selected for transmission by the second stage logic is selected for transmission by the third stage logic, the fifth tristate driver configured to receive the third data word and to drive the connection of the third stage logic with the third data word if the third data word is selected for transmission by the third stage logic.

21. The method of claim **14**, further comprising:
 receiving at a fourth tristate driver the data word selected by the selecting;
 selecting for transmission between the received data word and a third data word based upon a set of select signals;
 driving a second connection with the received data word via the fourth tristate driver if the received data word is selected by the selecting for transmission between the received data word and the third data word; and
 driving the second connection with the third data word if the third data word is selected by the selecting for transmission between the received data word and the third data word.

* * * * *

UNITED STATES PATENT AND TRADEMARK OFFICE
CERTIFICATE OF CORRECTION

PATENT NO. : 7,349,448 B2
APPLICATION NO. : 10/632885
DATED : March 25, 2008
INVENTOR(S) : Edward Chang

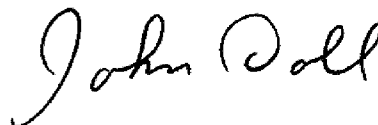
Page 1 of 1

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

On the face page, in field (56), under “Foreign Patent Documents”, in column 2, line 4, delete “GB 1483903 9/1974”.

Signed and Sealed this

Fourteenth Day of April, 2009

A handwritten signature in black ink that reads "John Doll". The signature is written in a cursive, flowing style.

JOHN DOLL

Acting Director of the United States Patent and Trademark Office