(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
9 December 2010 (09.12.2010)

PCT

(10) International Publication Number
WO 2010/141224 A1

(51) International Patent Classification:
G06F 15/78 (2006.01)

(21) International Application Number:
PCT/US2010/035455

(22) International Filing Date:
19 May 2010 (19.05.2010)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
12/478,450 4 June 2009 (04.06.2009) US

(71) Applicant *(for all designated States except US)*: **MICRON TECHNOLOGY, INC.** [US/US]; 8000 S. Federal Way, Boise, Idaho 83707-0006 (US).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: **WALKER, Robert** [US/US]; 9000 Deerland Grove Drive, Raleigh, North Carolina 27615 (US).

(74) Agents: **MANWARE, Robert A.** et al.; Fletcher Yoder PC, 7915 FM 1960 West, Suite 330, Houston, Texas 77070 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

Published:

— *with international search report (Art. 21(3))*

— *before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))*

(54) **Title**: DIRECT COMMUNICATION WITH A PROCESSOR INTERNAL TO A MEMORY DEVICE

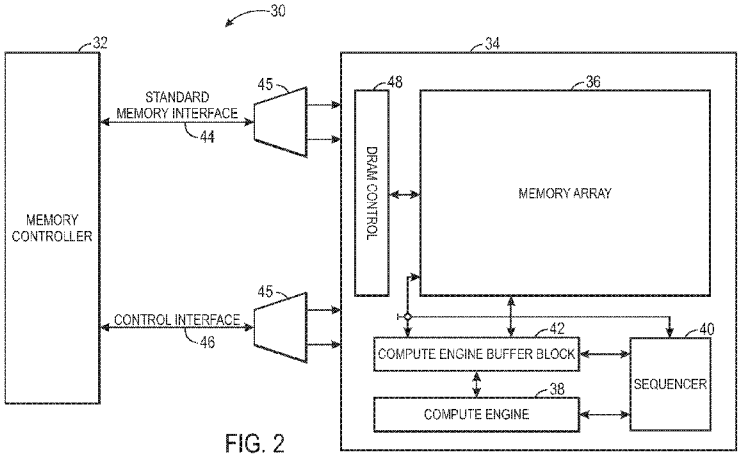

FIG. 2

(57) **Abstract**: Devices, systems, and methods of communicating information directly to a sequencer or a buffer in a memory device are provided. In some embodiments, instructions are sent directly from an external processor to a sequencer in the memory device, and the sequencer configures the instructions for an internal processor, such as one or more arithmetic logic units (ALUs) embedded on the memory device. Further, data to be operated on by the internal processor can be sent directly from the external processor to a buffer, and the sequencer can copy the data from the buffer to the internal processor. As power can be consumed each time a memory array is written to or read from, the direct communication of instructions and/or data can reduce the power consumed in writing to or reading from the memory array.

WO 2010/141224 A1

DIRECT COMMUNICATION WITH A PROCESSOR INTERNAL TO A MEMORY DEVICE

BACKGROUND

Field of Invention

[0001] Embodiments of the invention relate generally to systems, and more particularly, to systems with one or more processors internal to a memory device.

Description of Related Art

[0002] This section is intended to introduce the reader to various aspects of art that may be related to various aspects of the present invention, which are described and/or claimed below. This discussion is believed to be helpful in providing the reader with background information to facilitate a better understanding of the various aspects of the present invention. Accordingly, it should be understood that these statements are to be read in this light and not as admissions of prior art.

[0003] Electronic systems typically include one or more processors, which may retrieve and execute instructions, and store the results of the executed instructions to a suitable output or location. A processor generally includes arithmetic logic unit (ALU) circuitry, which is capable of executing instructions such as arithmetic and logic operations on one or more operands. For example, the ALU circuitry may add, subtract, multiply, or divide one operand from another, or may subject one or more operands to logic operations, such as AND, OR, XOR, and NOT logic functions. The various arithmetic and logic operations may have different degrees of complexity. For example, some operations may be executed

by inputting the operand(s) through the ALU circuitry in one cycle, while other operations may be require multiple clock cycles.

[0004] A number of components in the electronic system may be involved in directing a set of instructions to the ALU for execution. For example, the instructions and any corresponding data (e.g., the operands on which the operations will be executed) may be generated by a controller, or some other suitable processor in the electronic system. As the time or number of clock cycles required for the execution of a set of instructions may vary depending on the type of operation, the instructions and/or data may be written to a memory device, for example, a memory array, before being executed by the ALU. The instructions and data may be retrieved and sequenced and/or buffered before the ALU begins to execute the instructions on the data. To improve processing performance, the steps of writing, reading, sequencing, buffering, and executing instructions and/or data may be occurring substantially simultaneously on different instructions, or different parts of an instruction. This parallel processing may be referred to as "pipelining."

[0005] The process of executing instructions and the steps involved in pipelining instructions may consume power in the electronic system. For example, writing instructions and data to a memory array, and then retrieving the instructions for a processor to perform the operations may take energy. Furthermore, in some memory systems, an external bus may be required to input the instructions and data from a memory array to an ALU, and to output completed results back to the memory array or other suitable output. Such external data input/outputs may increase power consumption in the system.

[0006] In some memory systems, a processor may be embedded on a memory device, and may be referred to as a processor-in-memory (PIM). For example, one or more ALUs

may be embedded on a memory device, and may write to and read from a memory array without an external bus. However, the transfer of instructions and data in the memory device may still limit the efficiency of processing in an electronic system.

BRIEF DESCRIPTION OF DRAWINGS

[0007] Certain embodiments are described in the following detailed description and in reference to the drawings in which:

[0008] FIG. 1 depicts a block diagram of a processor-based system in accordance with an embodiment of the present technique;

[0009] FIG. 2 depicts a block diagram of a random access memory (RAM) system with embedded arithmetic logic units interfaced with a system on a chip (SOC) having a RAM controller, in accordance with an embodiment of the present technique;

[0010] FIG. 3 depicts a block diagram of a command path from the SOC to the DRAM array;

[0011] FIG. 4 depicts a block diagram of a command path from the SOC to the sequencer, in accordance with one or more embodiments of the present technique;

[0012] FIG. 5 depicts a block diagram of a data path from the SOC to the DRAM array; and

[0013] FIG. 6 depicts a block diagram of a data path from the SOC to either the compute engine buffer block, or the DRAM array, in accordance with one or more embodiments of the present technique.

DETAILED DESCRIPTION

[0014] Arithmetic logic unit (ALU) circuitry is generally used to process instructions in multiple stages. Processing the instructions may include executing the instructions, and storing the results of the executed instructions. More specifically, instructions, and the data on which the instructions will be executed, may be sent by a controller to the ALU, and may first be stored in a memory device to be retrieved when the ALU circuitry is available to execute the instructions. Once the instructions have been executed, the ALU may write the results of the operation to a memory component, or to any other suitable output.

[0015] In one or more embodiments of the present techniques, one or more processors, such as one or more ALUs, may be packaged with or embedded on a memory device. Such processors are hereinafter referred to as "internal processors." For example, the memory device may be a processor-in-memory (PIM), and may include embedded ALUs and a memory array, which may store instructions and data to be processed by the ALUs and the results from the completed instructions. Further, the memory device may include additional components such as a sequencer and buffer to configure and hold the instructions and/or data before the ALU executes the operations. As discussed, power may be consumed during the input of data and/or instructions from a memory component to an ALU and the output of the results from the ALU to the memory component. When the ALU is external to the memory component, an external input/output (I/O) may be employed. However, an external I/O may consume a significant amount of power from the memory system. For example, the high power consumption of an external I/O may be disadvantageous in a battery-run mobile electronic device, or in any electronic device operating on a power supply.

[0016] One embodiment of the present technique involves a memory device having an embedded processor, such as a compute engine, including one or more ALUs. As the compute engine is embedded on the memory device, no external bus may be required to communicate (e.g., transmit) information between the compute engine and the memory array. More specifically, instructions and/or data may be transmitted from the memory array to the compute engine without being routed outside (i.e., external to) the memory device. Similarly, results of completed operations may also be transmitted from the compute engine to the memory array within the memory device. A compute engine and a memory array embedded on the same memory device may eliminate the need for an external I/O between the two components, and may reduce power consumption for the electronic system.

[0017] In some electronic systems, an external processor, such as an integrated or discrete memory controller, may write data and instructions to the memory array on the memory device, and a sequencer may access the memory array to retrieve instructions while a buffer accesses the memory array to temporarily store data before the ALU is available to execute the instructions on the data. However, writing and reading information (e.g., instructions and/or data) to and from the memory array may consume power. For example, power may be consumed when the external processor writes information to the memory array, and further, when the information is retrieved for the ALU to perform the instructions.

[0018] One or more of the present techniques disclose systems and methods of sending instructions and/or data directly to a sequencer and/or buffer. In some embodiments, the instructions and/or data may be directly transmitted through an interface between an

external processor, such as a memory controller, and specific components of the memory device. As used herein, "directly" transmitting information may involve directly transmitting instructions to the sequencer without first writing the instructions to a memory array, and directly transmitting data to the buffer without first writing the data to the memory array. By directly communicating (e.g., sending) instructions and/or data to the sequencer or the buffer, the additional steps of writing and retrieving instructions and/or data to and from the memory array may be eliminated. As power is consumed each time the memory array is written to or read, such techniques may further reduce power consumption of the processor.

[0019] Several of the terms used above may be used throughout the present disclosure, and definitions of such terms are provided to facilitate a better understanding of the present techniques. A processor may be a compute engine, and may include one or more ALUs. As used herein, a compute engine may be a plurality of ALUs embedded on a memory device. The ALUs and a memory component, such as a memory array, may be on the same memory device. The compute engine may be controlled by a processor (e.g., a discrete controller or a controller integrated with a processor that performs other functions) not embedded on the memory device, and this processor may be referred to as an "external processor." The external processor may send "instructions," which refer to the task to be executed by the compute engine, which may also be referred to as "operations." For example, instructions may include arithmetic or logic operations, or any other task which an ALU may be suitable for performing. Instructions may sometimes require more than one "cycle," or more than one pass through one or more ALUs, before the instruction has been completely executed. "Operands" may refer to the data on which operations are executed, and depending on the instruction, one or more operands may be input into the

ALU circuitry at the same time. Additionally, in some operations, an operand may be input through an ALU in one cycle, and carried out of the ALU and back into the ALU as an operand for an additional cycle(s). An "interface" may refer to an operable coupling, such as an electrical connection, between a memory system having an embedded ALU and an external processor, such as an external memory controller. As will be explained, instructions, data, or other information may be transmitted through different types of interfaces between the memory system and the external processor.

[0020] Now turning to the figures, FIG. 1 depicts a processor-based system, generally designated by reference numeral 10. As is explained below, the system 10 may include various electronic devices manufactured in accordance with embodiments of the present technique. The system 10 may be any of a variety of types such as a computer, pager, cellular phone, personal organizer, control circuit, etc. In a typical processor-based system, one or more processors 12, such as a microprocessor, control the processing of system functions and requests in the system 10. As is explained below, the processor 12 and other subcomponents of the system 10 may include resistive memory devices manufactured in accordance with one or more embodiments of the present technique.

[0021] The system 10 typically includes a power supply 14. For instance, if the system 10 is a portable system, the power supply 14 may advantageously include a fuel cell, a power scavenging device, permanent batteries, replaceable batteries, and/or rechargeable batteries. The power supply 14 may also include an AC adapter, so the system 10 may be plugged into a wall outlet, for instance. The power supply 14 may also include a DC adapter such that the system 10 may be plugged into a vehicle cigarette lighter, for instance.

[0022] Various other devices may be coupled to the processor 12 depending on the functions that the system 10 performs. For instance, an input device 16 may be coupled to the processor 12. The input device 16 may include buttons, switches, a keyboard, a light pen, a mouse, a digitizer and stylus, and/or a voice recognition system, for instance. A display 18 may also be coupled to the processor 12. The input device 16 and/or the display 18 may each or both form a user interface. The display 18 may include an LCD, an SED display, a CRT display, a DLP display, a plasma display, an OLED display, LEDs, and/or an audio display, for example. Furthermore, an RF sub-system/baseband processor 20 may also be coupled to the processor 12. The RF sub-system/baseband processor 20 may include an antenna that is coupled to an RF receiver and to an RF transmitter (not shown). One or more communication ports 22 may also be coupled to the processor 12. The communication port 22 may be adapted to be coupled to one or more peripheral devices 24 such as a modem, a printer, a computer, or to a network, such as a local area network, remote area network, intranet, or the Internet, for instance.

[0023] The processor 12 generally controls the system 10 by executing instructions stored in the memory to implement software programs. The software programs may include an operating system, database software, drafting software, word processing software, and/or video, photo, or sound editing software, for example. The memory is operably coupled to the processor 12 to store and facilitate execution of various programs. For instance, the processor 12 may be coupled to the system memory 26, which may include dynamic random access memory (DRAM), and/or synchronous dynamic random access memory (SDRAM). The system memory 26 may include volatile memory, non-volatile memory, or a combination thereof. The system memory 26 is typically large so that it can store dynamically loaded instructions for applications and data.

[0024] The processor 12 may also be coupled to non-volatile memory 28, which is not to suggest that system memory 26 is necessarily volatile. The non-volatile memory 28 may include read-only memory (ROM), such as an EPROM, resistive read-only memory (RROM), and/or flash memory to be used in conjunction with the system memory 26. The size of the ROM is typically selected to be just large enough to store any necessary operating system, application programs, and fixed data. Additionally, the non-volatile memory 28 may include a high capacity memory such as a tape or disk drive memory, such as a hybrid-drive including resistive memory or other types of non-volatile solid-state memory, for instance.

[0025] One or more embodiments of the present techniques involve the communication between the processor 12 and components of the system memory 26. More specifically, the processor 12 may include a general purpose processor, a central processing unit, a processor core, an ASIC, a memory controller, and/or an ALU, for example, capable of sending and receiving signals from internal processors packaged with (e.g., embedded on) memory devices in the system memory 26. Components of the system 10 involved in the communication between the processor 12 and the components of the system memory 26 may be generally referred to as a "memory system" 30, as illustrated in the block diagram of FIG. 2. In some embodiments, a memory system 30 may include a memory device 34, which may be part of the system memory 26 of the system 10 (as in FIG. 1) and may have an internal processor. The memory system 30 may also include an external processor 32, which may be in a system-on-a-chip (SOC) with a more general purpose to collectively form a processor 12 of a processor-controlled system 10 (as in FIG. 1), for example. The external processor 32 may communicate with certain components of a memory device 34.

[0026] The memory system 30 may include components which have functions that are not limited to the communication between the external processor 32 and the memory device 34. For example, the external processor 32 may control devices in addition to the memory device 34. However, the external processor 32, as explained with respect to the memory system 30, may refer to one function of the external processor 32 which communicates with and/or controls certain components of the memory device 34. Likewise, not all parts of the system memory 26 may be part of the memory system 30. The "memory device" 34 may refer to components of the system memory 26 involved in the communication with the external processor 32, in accordance with the present techniques.

[0027] The external processor 32 and the memory device 34 may be operably coupled by a standard memory interface 44, which may allow data transfer between the external processor 32 and the memory device 34, and may allow the external processor 32 to send (e.g., transfer) commands to the memory device 34. In one or more embodiments, the types of standard memory interface 44 may include DDR, DDR2, DDR3, LPDDR, or LPDDR2, for example. Further, in some embodiments, an additional interface(s) may be configured to allow the transfer of data, and also commands (e.g., requests, grants, instructions, etc.), between the memory device 34 and the external processor 32. For example, the external processor 32 and the memory device 34 may also be operably coupled by a control interface 46, which may allow the transfer of commands between the external processor 32 and the memory device 34, including commands from the memory device 34 to the external processor 32. Both the standard memory interface 44 and the control interface 46 may include a multiplexer 45 to allow communication (e.g., data, commands, etc.) between the external processor 32 and different components of the

memory device 34 (e.g., the memory array 36, the sequencer 40, the compute engine buffer block 42, etc.)

[0028] The memory device 34 may include a memory array 36 and an internal processor, such as a compute engine 38. The memory array 36 may refer to any suitable form of storage, and may include, for example, a DRAM array or an SDRAM array. The external processor 32 may have access to the memory array 36, and may be able to send data or instructions to be executed by the compute engine 38. The compute engine 38 may include one or more arithmetic logic units (ALUs), or any other circuitry which may be capable of performing instructions sent from another processor (e.g., a memory controller), including, for example, arithmetic operations such as addition, subtraction, multiplication, and division, and logic functions such as AND, OR, XOR, and NOT functions.

[0029] The compute engine 38 may be embedded on the memory device 34 and capable of accessing the memory array 36, including retrieving information from, and storing information to the memory array 36. In some embodiments, the compute engine 38 may access information from the memory array via a sequencer 40 and compute engine buffer block 42. The sequencer 40 may organize (e.g., sequence) the instructions sent by the external processor 32 to the memory array 36 and store the data retrieved by the memory array 36 in the compute engine buffer block 42. Once the compute engine 38 has executed the instructions, the results may be stored in the compute engine buffer block 42 before they are written to the memory array 36. Further, as some instructions may require more than one clock cycle in the compute engine, intermediate results may also be stored in the compute engine buffer block 42. In some embodiments, the compute engine buffer block 42 may include more than one layer of buffers. For example, the buffer block 42 may

include a compute buffer, which may store operands, and/or an instruction buffer, which may store instructions. The buffer block 42 may also include additional buffers, such as a data buffer or a simple buffer, which may provide denser storage, and may store intermediate or final results of executed instructions.

[0030] In a typical memory system 30, an external processor 32 may write operands and instructions to the memory array 36 on the memory device 34. A sequencer 40 may access the memory array 36 to retrieve the instructions, and may copy the operands from the memory array 36 to the compute engine buffer block 42. For example, a depiction of how information may typically be written to a memory array 36 is provided in the block diagram of FIG. 3. The external processor 32 may send instructions through a standard memory interface 44 to the memory array 36, and a sequencer 40 accesses the memory array 36 to sequence the instructions for the compute engine 38. However, power is consumed when the external processor 32 writes the instructions to the memory array 36, and when the sequencer 40 reads the instructions from the memory array 36.

[0031] In one embodiment of the present techniques, power may be conserved by eliminating the step of transmitting instructions to and from the memory array 36. A block diagram depicting one embodiment is presented in FIG. 4, where the external processor 32 may transmit instructions directly to the sequencer. For example, instructions may be transmitted through either a standard memory interface 44, or through a control interface 46. As discussed, the control interface 46 may also allow the sequencer 40 or the compute engine 38 to directly communicate with the external processor 32. Writing to and reading from the memory array 36 may be reduced, as the sequencer 40 may not need to access the memory array 36 for each instruction to be sequenced. Rather, the sequencer 40 may

configure (i.e., sequence, organize, set up registers, etc.) the instructions for the compute buffer 38 when instructions are received directly from the external processor 32. As indicated by the bidirectional arrow between the compute engine 38 and the memory array 36, results of operations completed by the compute engine 38 may still be sent to the memory array 36. In some embodiments, instructions being transmitted into and out of the memory device 34 may first be sent to the compute engine buffer block 42 (as in FIG. 2). Furthermore, in other embodiments, the instructions may be sent to different components of the memory device 34 depending on an application, or the instructions to be executed. Instructions may be communicated (e.g., transferred, transmitted or otherwise sent) directly to any of the sequencer 40, the memory array 36, or the compute engine buffer block 42, and the signal carrying the instructions may also carry an identifier, additional bits, or some other indicator to indicate that the instructions are targeted for a particular component of the memory device 34.

[0032] One or more of the present techniques can also include systems and methods of sending data from an external processor 32 directly to a compute engine buffer block 42. In a typical memory system, a processor may write data to a memory array before the data is retrieved for operation. An illustration of a data path from an external processor 32 to a memory array 36 is provided in the block diagram of FIG. 5. The data 50, shown as data A and B, may be operands to be executed (e.g., operated on) by the compute engine 38. Before the compute engine 38 may operate on the data 50, a sequencer 40 (as in FIG. 2, not shown in FIG. 5) may copy the data 50 from the memory array 36 to the compute engine buffer block 42. The data 50 may then be maneuvered in the buffer block 42 to be operated on by the compute engine 38.

[0033] In one embodiment, data writing to the memory array 36 and the data reading from the memory array 36 may be reduced, which may conserve the power required to write to and read from the memory array 36. As illustrated in the block diagram of FIG. 6, the external processor 32 may send (e.g., write) data directly to the compute engine buffer block 42 via the standard memory interface 44. Though the external processor 32 and the buffer block 42 may still have access to the memory array 36, certain data 50 may be directly sent to the buffer block 42, such as to reduce the power consumed in writing and reading to and from the memory array 36. Thus, a sequencer 40 (as in FIG. 2, not shown in FIG. 6), may read data from the compute engine buffer block 42 to be operated on by the compute engine 38.

[0034] As discussed, in some embodiments, the data 50 may be sent to different components of the memory device 34 depending on an application, or the data to be operated on. The data 50 may be sent directly to either the memory array 36, the compute engine buffer block 42, or some other component on the memory device 34, and the data signal may also carry an identifier, additional bits, or some other indicator to indicate that the data 50 is targeted for a particular component of the memory device 34.

[0035] While the invention may be susceptible to various modifications and alternative forms, specific embodiments have been shown by way of example in the drawings and have been described in detail herein. However, it should be understood that the invention is not intended to be limited to the particular forms disclosed. Rather, the invention is to cover all modifications, equivalents, and alternatives falling within the spirit and scope of the invention as defined by the following appended claims.

CLAIMS

1. A system comprising:

a memory device comprising:

a buffer configured to receive data directly from an external processor;

a sequencer configured to receive instructions directly from the external processor; and

an internal processor, wherein

the internal processor is configured to process the instructions received by the sequencer on the data received by the buffer; and

an external processor configured to communicate directly with one or more of the buffer or the sequencer.

2. The system, as set forth in claim 1, wherein the memory device comprises a memory array configured to receive data from the external processor.

3. The system, as set forth in claim 1, wherein the system comprises a memory interface configured such that the external processor can send data to both the buffer and a memory array in the memory device.

4. The system, as set forth in claim 3, wherein the memory interface comprises a multiplexer configured to enable the external processor to send data to one or more of the buffer and a memory array.

5. The system, as set forth in claim 1, wherein the sequencer is configured to receive instructions directly from the external processor via a control interface.

6. The system, as set forth in claim 1, wherein the sequencer is configured to receive a signal directly from the external processor via a memory interface, wherein the signal received via the memory interface comprises instructions and an indicator configured to indicate that the signal is targeted for the sequencer.

7. The system, as set forth in claim 1, wherein the external processor is configured to communicate with both the buffer and the sequencer via a control interface.

8. The system, as set forth in claim 1, wherein the external processor is configured to communicate with each of the buffer, the sequencer, and a memory array by sending a signal via a memory interface, wherein the signal comprises an indicator to indicate whether the signal is targeted for the buffer, the sequencer, or the memory array.

9. The system, as set forth in claim 1, wherein the buffer comprises a block of one or more layers of buffers.

10. The system, as set forth in claim 1, wherein the external processor is a discrete memory controller or a processor with an integrated memory controller.

11. The system, as set forth in claim 1, wherein the internal processor is packaged in the memory device.

12. The memory system, as set forth in claim 1, wherein the internal processor is embedded on the memory device.

13. A system comprising:

a memory device comprising:

a buffer configured to receive data directly from an external processor; and

an internal processor configured to process the data received by the buffer; and

the external processor, configured to communicate directly with the buffer.

14. The system, as set forth in claim 13, comprising an interface operably coupling the external processor and the buffer.

15. The system, as set forth in claim 13, wherein the external processor is configured to communicate with one or more of the internal processor and a sequencer on the memory device.

16. A system comprising:

a memory device comprising:

a sequencer configured to receive instructions directly from an external processor; and

an internal processor configured to process the instructions received by the sequencer; and

the external processor, configured to communicate directly with the sequencer.

17. The system, as set forth in claim 16, wherein the memory device comprises a memory component, and wherein results of the processed instructions are written to the memory component.

18. The system, as set forth in claim 16, wherein the internal processor comprises one or more arithmetic logic units (ALUs), and wherein the ALUs are embedded on the memory device.

19. A method of operating a memory device, comprising:

receiving instructions at a sequencer in the memory device, wherein the instructions are received directly from an external processor;

configuring the instructions for an internal processor of the memory device, wherein the instructions are configured by the sequencer;

receiving data at a buffer in the memory device, wherein the data is received directly from the external processor; and

processing the instructions on the data, wherein the instructions are processed by the internal processor.

20. The method, as set forth in claim 19, wherein the interface is configured to enable direct communication between the external processor and one or more of the sequencer, the buffer, or a memory array in the memory device.

21. The method, as set forth in claim 19, wherein configuring the instructions comprises setting registers and signals for the internal processor to complete the instructions.

22. The method, as set forth in claim 19, wherein executing the instructions comprises performing arithmetic and/or logic operations.

23. The method, as set forth in claim 19, comprising sending the instructions from the external processor to the sequencer.

24. The method, as set forth in claim 19, comprising sending the data from the external processor to the buffer.

25. The method, as set forth in claim 19, wherein the internal processor comprises one or more arithmetic logic units.

26. A method of controlling a memory device having an embedded arithmetic logic unit (ALU), comprising:

sending a signal comprising data to a compute engine buffer block on the memory device;

inputting the data to the ALU as one or more operands; and

sending a signal comprising instructions directly from a memory controller to a sequencer on the memory device, such that the ALU uses the signal to execute the instructions on the one or more operands.

27. The method, as set forth in claim 26, wherein the signal comprising data comprises an indicator to direct the data to the compute engine buffer block.

28. The method, as set forth in claim 26, wherein the signal comprising instructions comprises an indicator to direct the instructions to the sequencer.

29. The method, as set forth in claim 26, comprising:

generating results based on the instructions executed by the ALU; and

storing the results in a memory array on the memory device.

30. The method, as set forth in claim 26, comprising:

generating results based on the instructions executed by the ALU; and

storing the results in the compute engine buffer block.

31. The method, as set forth in claim 26, comprising:

generating results based on the instructions executed by the ALU; and

outputting the results from the memory device.

32. The method, as set forth in claim 26, wherein the signal comprising instructions is sent through a first interface between the memory controller and the sequencer, and the signal comprising data is sent through either the first interface or a second interface between the memory controller and the compute engine buffer block.

33. A memory device comprising:

a sequencer configured to receive instructions directly from an external processor; and

an internal processor configured to execute the instructions received by the sequencer.

34. The memory device, as set forth in claim 33, comprising a buffer configured to receive data directly from the external processor.

35. The memory device, as set forth in claim 34, wherein the internal processor is configured to execute the instructions on the data received by the buffer.

36. The memory device, as set forth in claim 34, comprising an external input/output (I/O) operably coupled to the buffer and the sequencer and configured to receive the data and the instructions from the external processor.

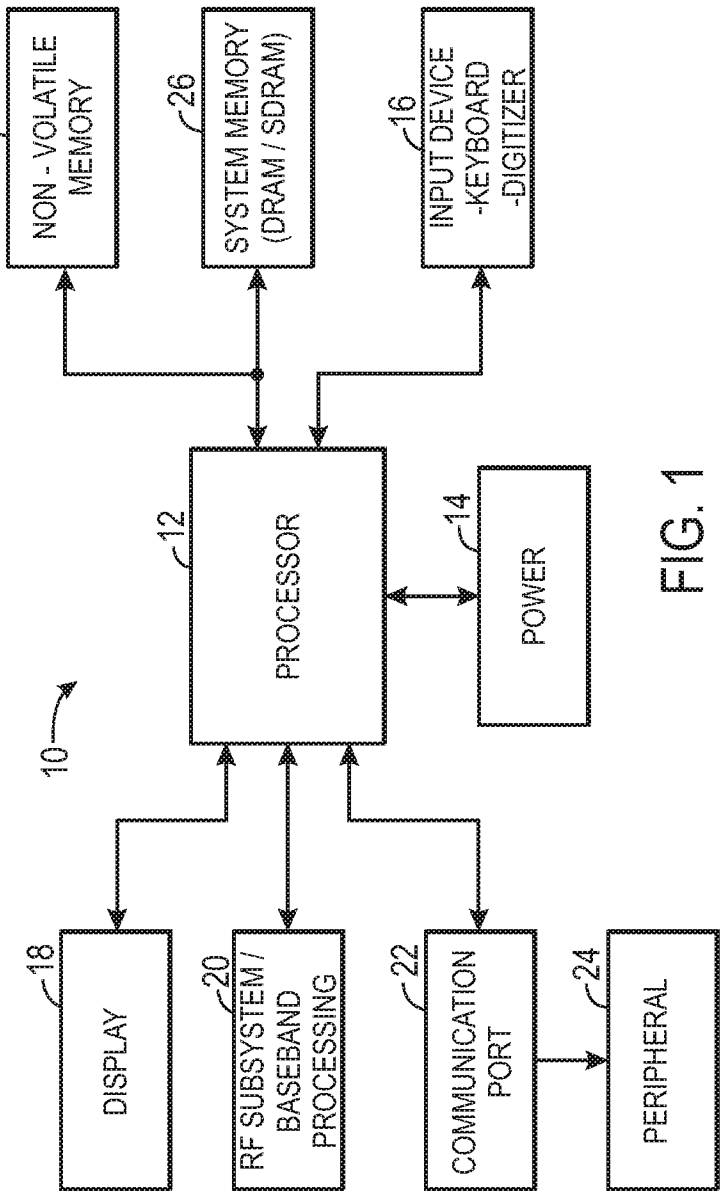
10
12
PROCESSOR
14
POWER
16
INPUT DEVICE
-KEYBOARD
-DIGITIZER
18
DISPLAY
20
RF SUBSYSTEM /
BASEBAND
PROCESSING
22
COMMUNICATION
PORT
24
PERIPHERAL
26
SYSTEM MEMORY
(DRAM / SDRAM)
28
NON - VOLATILE
MEMORY

FIG. 1

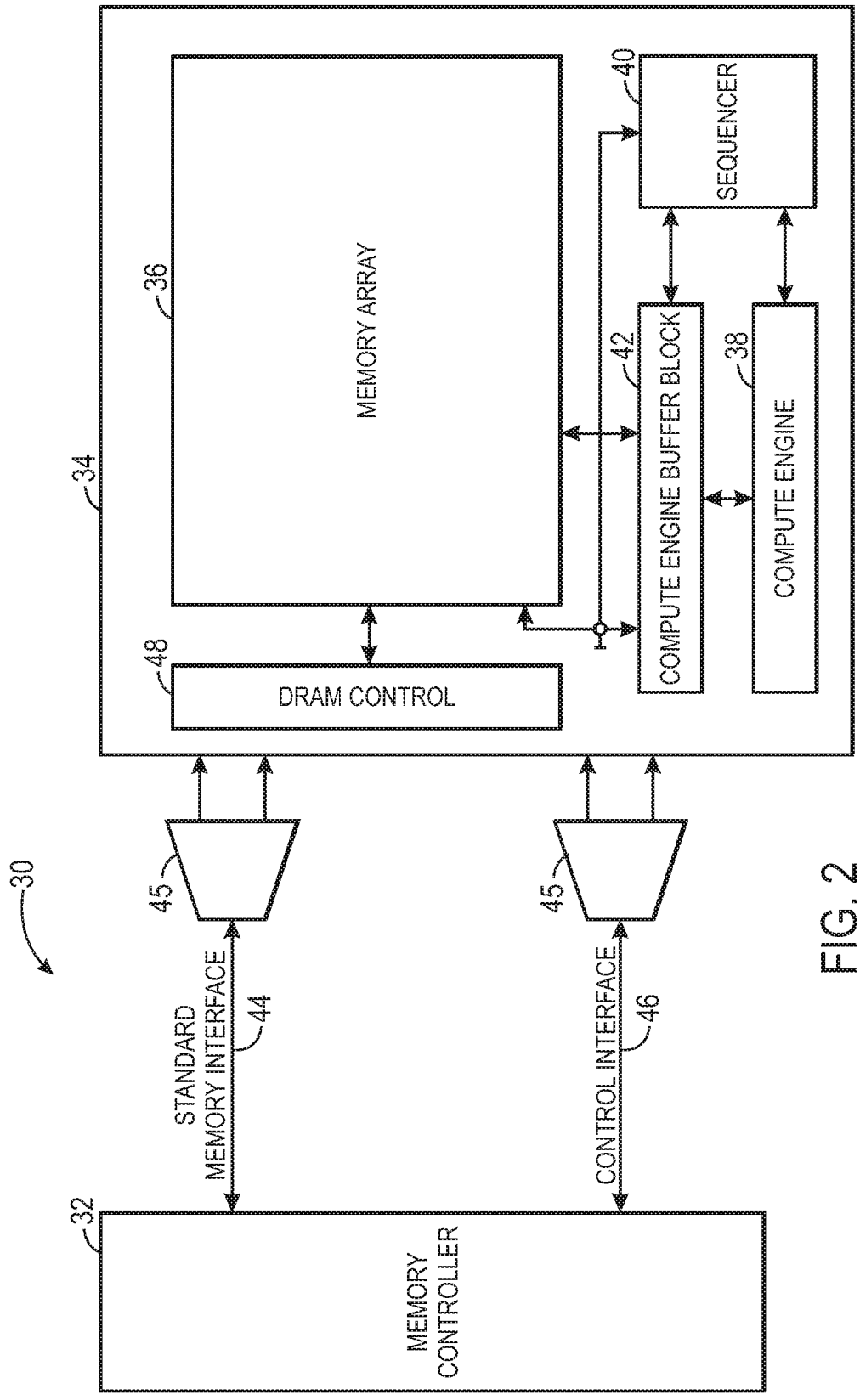
30
34
36
MEMORY ARRAY
40
SEQUENCER
42
COMPUTE ENGINE BUFFER BLOCK
38
COMPUTE ENGINE
48
DRAM CONTROL
45
45
STANDARD MEMORY INTERFACE
44
CONTROL INTERFACE
46
32
MEMORY CONTROLLER

FIG. 2

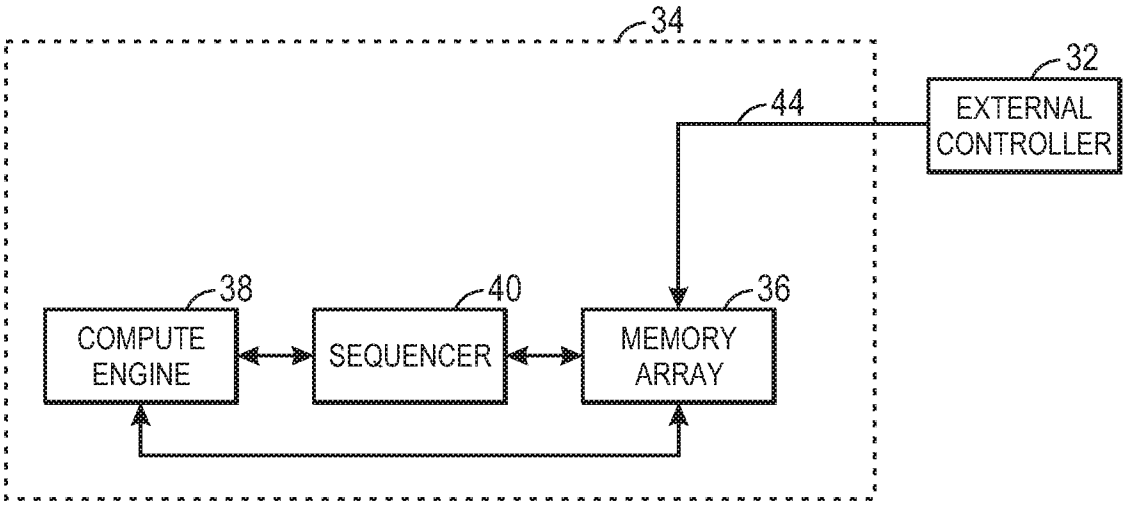
34
32
44
EXTERNAL CONTROLLER
38
40
36
COMPUTE ENGINE
SEQUENCER
MEMORY ARRAY

FIG. 3

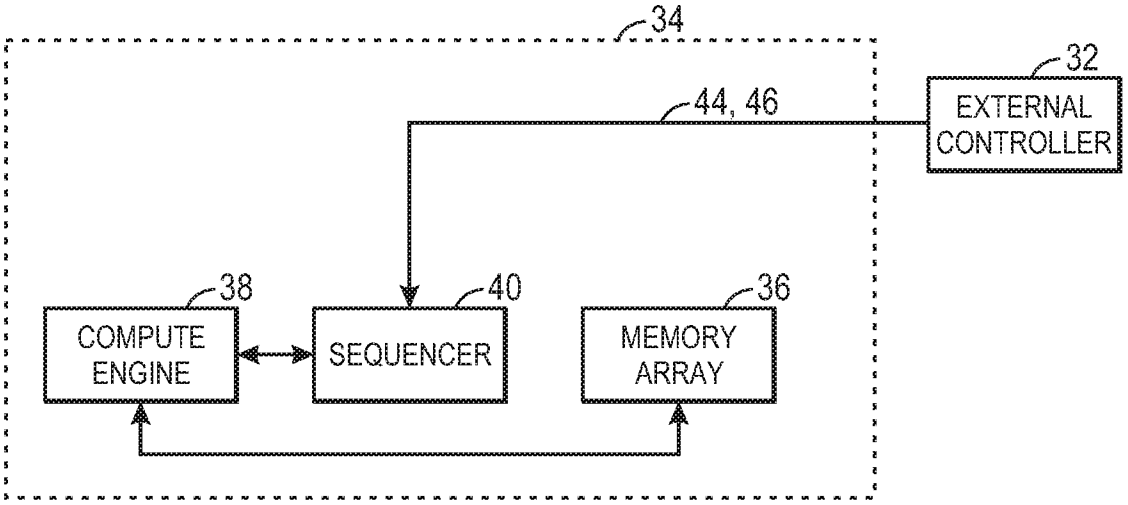
34
32
44, 46
EXTERNAL CONTROLLER
38
40
36
COMPUTE ENGINE
SEQUENCER
MEMORY ARRAY

FIG. 4

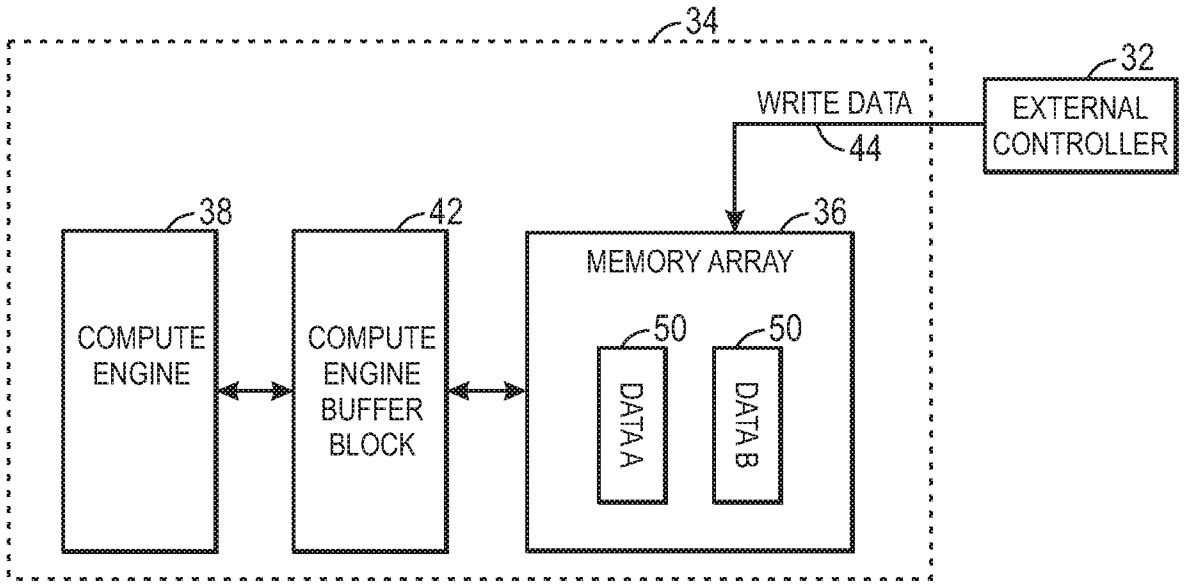
34
32
WRITE DATA
44
EXTERNAL CONTROLLER
38
42
36
COMPUTE ENGINE
COMPUTE ENGINE BUFFER BLOCK
MEMORY ARRAY
50
50
DATA A
DATA B

FIG. 5

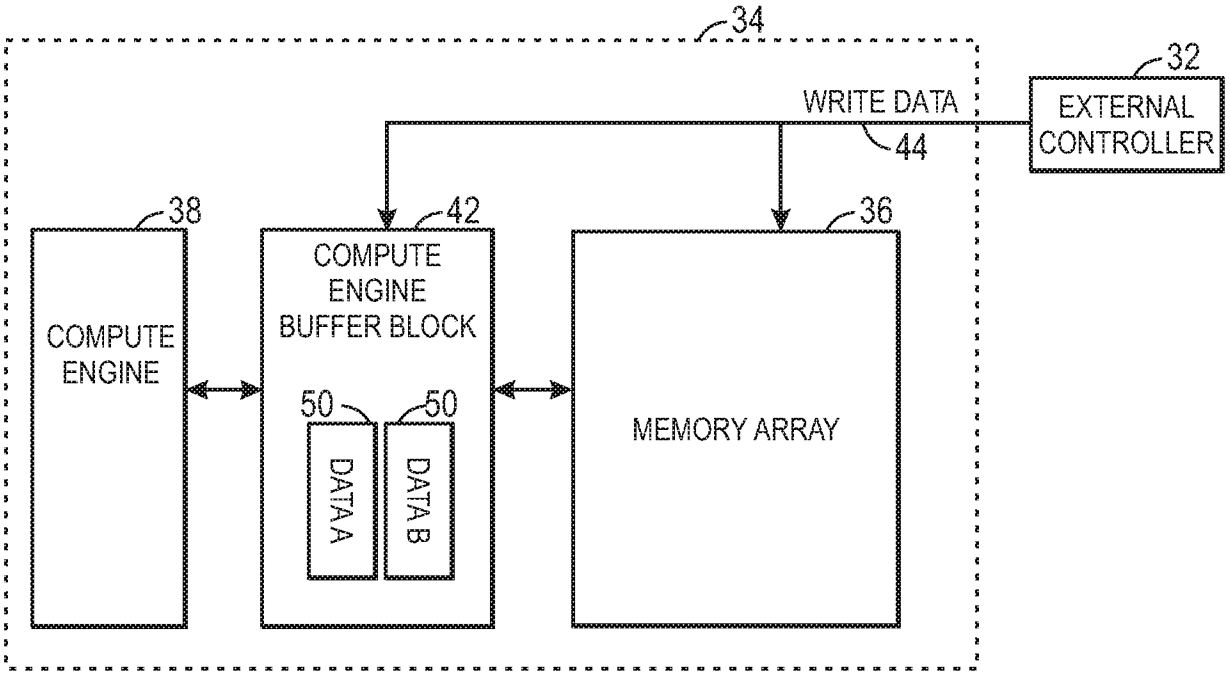
34
32
WRITE DATA
44
EXTERNAL CONTROLLER
38
42
36
COMPUTE ENGINE
COMPUTE ENGINE BUFFER BLOCK
MEMORY ARRAY
50
50
DATA A
DATA B

FIG. 6

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2010/035455

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F15/78
ADD.

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)
EPO-Internal

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2004/215852 A1 (KLEIN DEAN A [US]) 28 October 2004 (2004-10-28) figure 1 paragraph [0016] - paragraph [0018] paragraph [0032] paragraph [0009] paragraph [0005] paragraph [0022] paragraph [0025] ----- -/--	1-36

[X] Further documents are listed in the continuation of Box C.

[X] See patent family annex.

* Special categories of cited documents :

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier document but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art.

"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
15 November 2010	24/11/2010

Name and mailing address of the ISA/	Authorized officer
European Patent Office, P.B. 5818 Patentlaan 2 NL - 2280 HV Rijswijk Tel. (+31-70) 340-2040, Fax: (+31-70) 340-3016	Bosch Vivancos, P

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2010/035455

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	ELLIOTT D G ET AL: "COMPUTATIONAL RAM: IMPLEMENTING PROCESSORS IN MEMORY", IEEE DESIGN & TEST OF COMPUTERS, IEEE SERVICE CENTER, NEW YORK, NY, US, vol. 16, no. 1, 1 January 1999 (1999-01-01), pages 32-41, XP000823403, ISSN: 0740-7475, DOI: DOI:10.1109/54.748803 the whole document	1-36

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No
PCT/US2010/035455

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2004215852 A1	28-10-2004	US 2006206641 A1	14-09-2006