



- (51) **International Patent Classification:**
G06F 12/10 (2006.01) *G11C 7/00* (2006.01)
- (21) **International Application Number:**
PCT/US2013/021168
- (22) **International Filing Date:**
11 January 2013 (11.01.2013)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (71) **Applicant:** EMPIRE TECHNOLOGY DEVELOPMENT LLC [US/US]; 2711 Centerville Road, Suite 400, Wilmington, DE 19808 (US).
- (72) **Inventor:** MA, Xudong; 318 North Lincoln Avenue, Apt. 20, Monterey Park, CA 91755 (US).
- (74) **Agent:** HOPE, Leonard J.; Hope Baldauff Hartman, LLC, 1720 Peachtree Street, N.W., Suite 1010, Atlanta, GA 30309 (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, BN, 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, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, 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, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, 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, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

Published:

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

(54) **Title:** PAGE ALLOCATION FOR FLASH MEMORIES

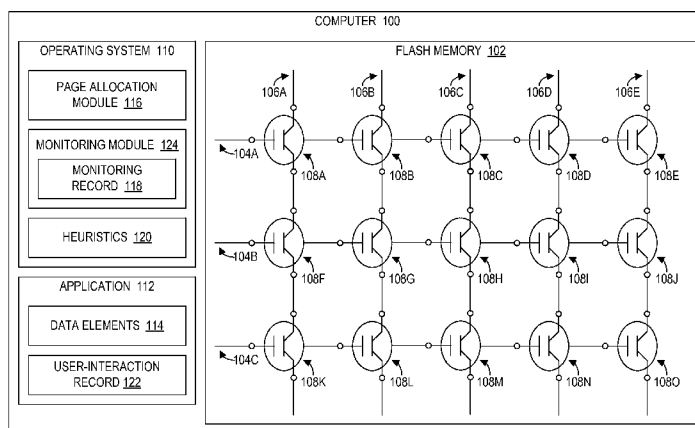


Fig. 1

(57) **Abstract:** Technologies are described herein for allocating pages in a flash memory. Some example technologies may receive multiple data elements and a write request to write the multiple data elements to the flash memory. Example technologies may identify a correlation between a subset of the data elements based on correlation criteria. Example technologies may allocate neighboring pages of the flash memory for storing the subset of the data elements. Example technologies may write the subset of the data elements into the allocated pages.

PAGE ALLOCATION FOR FLASH MEMORIES**BACKGROUND**

[0001] Unless otherwise indicated herein, the materials described in this section are not prior art to the claims in this application and are not admitted to be prior art by inclusion in this section.

[0002] In a flash memory, information can be recorded by altering the threshold voltage level of one or more memory cells within the flash memory. During a write operation, modification of the threshold voltage level of a first memory cell may cause disturbances to the threshold voltage level of a nearby second memory cell. That is, the threshold voltage level of the second memory cell may unintentionally change, which can result in an error when decoding the second memory cell. Such disturbances may be referred to as inter-cell interference.

[0003] Some approaches to addressing decoding errors caused by inter-cell interference involve utilization of joint decoding decisions. In a joint decoding decision, the threshold voltage levels of a requested memory cell as well as non-requested, neighboring memory cells may be read. The joint decoding decision may utilize the threshold voltage levels of the neighboring memory cells in order to accurately decode the requested memory cell. In particular, the threshold voltage levels of the neighboring memory cells may be utilized to compensate for potential interference to the requested memory cell. However, the implementation of a joint decoding decision may result in inefficiencies in systems that utilize conventional page allocation schemes for flash memories.

SUMMARY

[0004] The present disclosure generally describes techniques for allocating pages in a flash memory. Some example methods may receive multiple data elements and a write request to write the multiple data elements to the flash memory. Example methods may identify a correlation between a subset of the data elements based on correlation criteria. Example methods may allocate neighboring pages of the flash memory for storing the subset of the data elements. Example methods may write the subset of the data elements into the allocated pages.

[0005] The present disclosure generally further describes some computer-readable medium having computer-executable instructions stored thereon which, when executed by a computer, cause the computer to perform one or more operations for allocating pages in a flash memory. Some example operations may include receiving multiple data elements and a write request to write the multiple data elements to the flash memory. Example operations

may include identifying a correlation between a subset of the data elements based on correlation criteria. Example operations may include allocating neighboring pages of the flash memory for storing the subset of the data elements. Example operations may include writing the subset of the data elements into the allocated pages.

[0006] The present disclosure generally also describes some computing devices adapted to allocate pages in a flash memory. The computing devices may include a flash memory and an operating system. The flash memory may include a multiple memory cells arranged to form multiple memory pages. The operating system may be coupled to the flash memory and may be configured to perform one or more operations. Some example operations may include receiving multiple data elements and a write request to write the multiple data elements to the flash memory. Example operations may include identifying a correlation between a subset of the data elements based on correlation criteria. Example operations may include allocating neighboring pages of the flash memory for storing the subset of the data elements. Example operations may include writing the subset of the data elements into the allocated pages.

[0007] The foregoing Summary is illustrative only and is not intended to be in any way limiting. In addition to the illustrative aspects, embodiments, and features described above, further aspects, embodiments, and features will become apparent by reference to the Figures and the following Detailed Description.

DESCRIPTION OF THE FIGURES

[0008] The foregoing and other features of this disclosure will become more fully apparent from the following Detailed Description, accompanying Figures, and appended claims. Understanding that these Figures depict only several embodiments in accordance with the disclosure and are, therefore, not to be considered limiting of its scope, the disclosure will be described with additional specificity and detail with reference to the accompanying Figures, in which:

FIG. 1 is a diagram illustrating an example architecture of a computer adapted to allocate pages in a flash memory;

FIG. 2 is a diagram illustrating an example arrangement of pages in a memory block of the flash memory;

FIG. 3 is a flow diagram illustrating an example process adapted to allocate pages in a flash memory;

FIG. 4 is a block diagram illustrating a computer hardware architecture for an example computing system; and

FIG. 5 is a schematic diagram illustrating a computer program product that includes a computer program for executing a computer process on a computing device;

all arranged in accordance with at least some embodiments presented herein.

DETAILED DESCRIPTION

[0009] In the present Detailed Description, reference is made to the accompanying Figures, which form a part hereof. In the Figures, similar symbols typically identify similar components, unless context dictates otherwise. The illustrative embodiments described in the Detailed Description and Figures are not meant to be limiting. Other embodiments may be utilized, and other changes may be made, without departing from the spirit or scope of the subject matter presented herein. It will be readily understood that the aspects of the present disclosure, as generally described herein, and illustrated in the Figures, can be arranged, substituted, combined, separated, and designed in a wide variety of different configurations, all of which are explicitly contemplated herein.

[0010] This disclosure is generally drawn, *inter alia*, to technologies adapted to implement page allocation schemes for flash memories. The technologies may be adapted to identify highly correlated files, file segments, or other data elements. As used herein, highly correlated data elements may refer to data elements that are accessed concurrently, sequentially, and/or within a threshold amount of time. The technologies may be adapted to allocate neighboring memory pages in a flash memory for storing the highly correlated data elements. The highly correlated data elements may be written into the allocated neighboring memory pages.

[0011] A common drawback experienced in flash memories is inter-cell interference. Inter-cell interference may refer to disturbances to nearby memory cells that result when writing to an intended memory cell. As flash memory technology continues to evolve, the geometrical sizes of each memory cell in a flash memory may continue to scale down, and the memory cells may be arranged closer together. These advancements may allow for the placement of more memory cells within the same area on the flash memory. As a result, inter-cell interference may become worse in future flash memories.

[0012] Some approaches to addressing inter-cell interference may involve the utilization of a joint decoding decision. In conventional bit-wise decoding processes, each requested memory cell may be decoded independently. However, the decoded result obtained from the requested memory cell may be incorrect due to inter-cell interference. In a joint decoding decision, neighboring memory cells may be utilized in order to more accurately decode the requested memory cell. Neighboring memory cells are memory cells that are geometrically close to one another. Two memory pages are called neighbors, or neighboring memory pages, if at least one memory cell in one memory page is a neighbor to at least one memory cell in the other memory page. Neighboring memory cells may include memory

cells located on the same page as the requested memory cell as well as memory cells located on neighboring pages.

[0013] Conventional flash memories may allocate pages as they become available. That is, a conventional flash memory may simply allocate the next clean page when a page is requested, without regard to characteristics of the data elements to be stored on the flash memory. The next clean page may not be neighboring or even close to the previous page. As a result, highly correlated data that spans multiple pages may be stored on non-neighboring or distant pages.

[0014] When implementing a joint decoding decision, the reading of each requested memory cell corresponding to the highly correlated data elements may include the reading of non-requested neighboring memory cells. The neighboring memory cells may be located in pages separate from the pages that contain the requested memory cells. The reading of multiple pages that do not contain requested memory cells may result in various performance inefficiencies, in terms of overall reading speeds, error rates, implementation complexities, or combinations thereof.

[0015] Various embodiments described herein provide for the allocation of neighboring pages for storing highly correlated data elements. By storing the highly correlated data elements on neighboring pages, the quantity of pages that do not contain requested memory cells may be substantially decreased when implementing a joint decoding decision. For example, a first read request and a second read request may request highly correlated data. A given page that is read may include a memory cell that is not part of the first read request but is part of a joint decoding decision for a memory cell on another page. However, the given page may also include a memory cell (which may or may not be the same) that is part of the second read request. As a result, the allocation of neighboring pages for storing highly correlated data elements may result in improved performance efficiencies over conventional page allocation approaches when implementing a joint decoding decision.

[0016] FIG. 1 is a diagram illustrating an example architecture of a computer 100 adapted to allocate pages in a flash memory, arranged in accordance with at least some embodiments presented herein. The computer 100 may include a flash memory 102. The flash memory 102 may be an internal memory device or an external memory device with respect to the computer 100. The flash memory 102 might be a NOT AND (“NAND”) type flash memory or a NOT OR (“NOR”) type flash memory. The flash memory 102 might also be a single-level-cell type flash memory or a multi-level-cell type flash memory. Other types of flash memory devices might also be utilized.

[0017] A portion of an example internal configuration of the flash memory 102 is illustrated in FIG. 1. The flash memory 102 may include multiple word lines, including a first wordline 104A, a second wordline 104B, and a third wordline 104C (collective referred

to as wordlines 104). The wordlines 104 are sometimes colloquially referred to as “rows” in the flash memory 102. The flash memory 102 may further include multiple bitlines, including a first bitline 106A, a second bitline 106B, a third bitline 106C, a fourth bitline 106D, and a fifth bitline 106E (collectively referred to as bitlines 106). The bitlines 106 are sometimes colloquially referred to as “columns” in the flash memory 102.

[0018] The flash memory 102 may also include multiple memory cells 108A-108O (collectively referred to as memory cells 108). Each of the memory cells 108 may be arranged at the intersection of one of the wordlines 104 and one of the bitlines 106. Each of the memory cells 108 may include a channel that is arranged between a source and a drain. A floating gate may be arranged over the channel and may be electrically separated from the channel. A control gate may be arranged over the floating gate, and may be electrically separated from the floating gate. The control gate may be coupled to one of the wordlines 104, and the source and drain may be coupled to one of the bitlines 106. Control gates from multiple memory cells may share a given wordline, and sources and drains from multiple memory cells may share a given bitline. For example, the control gates of the memory cells 108A-108E may share the first wordline 104A, and the sources and drains of the memory cells 108A, 108F, 108K may share the first bitline 106A.

[0019] Each memory cell may be programmed by storing a charge on the floating gate. A charged floating gate may represent a first logic state (e.g., a logical “0”), and a non-charged floating gate may represent a second logic state (e.g., a logical “1”). A memory cell may be programmed by applying a first voltage (e.g., a negative voltage) to a corresponding bitline and applying a second voltage (e.g., a positive voltage) that is higher than the first voltage to a corresponding wordline. Through a process known as tunnel injection, the floating gate corresponding to the intersecting bitline and wordline may accumulate and trap a negative charge. The accumulated negative charge in the floating gate may effectively increase the threshold voltage level of the memory cell. Thus, by reading the threshold voltage level of a given memory cell, the logic state programmed into the memory cell can be decoded by determining whether the threshold voltage level has increased.

[0020] Due to parasitic capacitance-coupling effects, the application of a voltage to an intended memory cell may unintentionally affect the voltage levels of control gates, sources, and/or drains of neighboring memory cells. The neighboring memory cells may include memory cells that are immediately adjacent to the intended memory cell and/or memory cells that are within some threshold distance to the intended memory cell. The unintentional effect on neighboring memory cells caused by writing to an intended memory cell may be referred to as inter-cell interference. For example, writing to the memory cell 108H may unintentionally affect the voltage levels of one or more of the memory cells 108B-108D, 108G, 108I, 108L-108N. When the threshold voltage level of an unintentionally

affected memory cell is read, the decoded logic state of the memory cell may be incorrect due to the inter-cell interference (e.g., a memory cell that should be decode as a logical “1” may be incorrectly decoded as a logical “0”).

[0021] A page may refer to the smallest area of the flash memory 102 that can support a read operation or a write operation. A block may refer to the smallest area of the flash memory 102 that can support an erase operation. A page may include multiple memory cells. A block may include multiple pages. Each of the wordlines 104 may include one or more pages. In some embodiments, each of the wordlines 104 may correspond to one page. In some other embodiments, each of the wordlines 104 may correspond to two or more pages. In some other embodiments, where each memory cell may store multiple bits, a page may refer to bits stored in a group of memory cells. For example, one memory page may refer to the most significant bits of one group of memory cells. In this case, the memory page may be considered to contain the group of memory cells.

[0022] The computer 100 may be configured to execute an operating system 110 and an application 112. The application 112 may be configured to generate a read request for reading data elements from the flash memory 102 or a write request for writing data elements to the flash memory 102. The operating system 110 may be configured to perform read operations on the flash memory 102 to fulfill the read request. The operating system 110 may also be configured to perform write operations on the flash memory 102 to fulfill the write request.

[0023] The operating system 110 may be configured to implement a joint decoding decision when reading data elements from the flash memory 102. When implementing a joint decoding decision, the operating system 110 may read the threshold voltage levels of not only a requested memory cell but also memory cells neighboring the requested memory cell. The joint decoding decision may decode the requested memory cell utilizing the threshold voltage levels of the neighboring memory cells in order to compensate for possible interference caused by the neighboring memory cells.

[0024] When the operating system 110 receives a write request to write data elements 114 to the flash memory 102, a page allocation module 116 may be configured to determine correlations between two or more of the data elements 114. The correlations may be determined utilizing a monitoring record 118, heuristics 120, and/or a user-interaction record 122, as described in greater detail below with reference to FIG. 3. The correlations may be utilized to form groups of two or more highly correlated data elements. The page allocation module 116 may be configured to allocate neighboring pages of the flash memory 102 for storing each group of highly correlated data elements. The page allocation module 116 may be implemented in the operating system 110, a flash memory controller (not shown),

or some other software, hardware, or firmware capable of implementing the page allocation technologies described herein.

[0025] In an illustrative example, the operating system 110 may receive a series of read requests that identify multiple memory cells to be read from the flash memory 102. When implementing a joint decoding decision, the operating system 110 may be configured to read not only the requested memory cells, but also non-requested memory cells that are neighboring the requested memory cells. Some of the neighboring memory cells may be contained on the same pages as the requested memory cells. However, some other neighboring memory cells may be contained in neighboring pages that do not contain the requested memory cells identified in the series of read requests. The reading of multiple neighboring pages that do not contain the requested memory cells may result in various performance inefficiencies, in terms of overall reading speeds, error rates, implementation complexities, or combinations thereof.

[0026] In order to reduce the quantity of unique pages that are read but do not contain the requested memory cells, the page allocation module 116 may be configured to allocate neighboring pages for highly correlated data elements. Highly correlated data elements may refer to data elements that are accessed concurrently, sequentially, or within a threshold amount of time. For example, highly correlated data elements may refer to data elements that have some relation such that the data elements are likely to be accessed together via the series of read requests. When highly correlated data elements have been identified, the page allocation module 116 may be configured to allocate neighboring pages in the flash memory 102 for storing the highly correlated data elements. When the neighboring pages in the flash memory 102 have been allocated, the operating system 110 may write the highly correlated data elements into the allocated neighboring pages.

[0027] When the operating system 110 receives the series of read requests, the operating system 110 may be configured to read, for each requested memory cell, a page containing the requested memory cell as well as neighboring pages containing non-requested memory cells neighboring the requested memory cell. The series of read requests may be directed to reading the highly correlated data elements stored in the flash memory 102. As a result, at least some of the neighboring pages that are read for one read request may contain a requested memory cell for another read request. Thus, by storing highly correlated data on neighboring pages, a greater quantity of pages may contain both requested memory cells and non-requested neighboring memory cells for satisfying the series of read requests. Therefore, the quantity of unique pages that are read but do not contain a requested memory cell identified in the series of read requests can be substantially reduced. In some embodiments, the read operations of the operating system 110 may be optimized such that overlapping pages that are read between different read requests are read only once.

[0028] FIG. 2 is a diagram illustrating an example arrangement of pages in a memory block 200 of the flash memory 102, arranged in accordance with at least some embodiments presented herein. The memory block 200 may include multiple pages, including a first page 202A, a second page 202B, a third page 202C, a fourth page 202D, a fifth page 202E, and a sixth page 202F (collectively referred to as pages 202). The pages 202 may represent adjacent pages in the flash memory 102. In some embodiments, each of the pages 202 may represent one of the wordlines 104 in the flash memory 102. In some other embodiments, two or more of the pages 202 may be included in one of the wordlines 104 in the flash memory 102.

[0029] Conventional page allocation mechanisms may allocate the next clean page available in the flash memory 102 in response to a write request, without regard to characteristics of the data elements to be stored in the flash memory. For example, in conventional page allocation mechanisms, highly correlated data elements may be stored in the first page 202A, the third page 202C, and the fifth page 202E. In this example, the operating system 110 may receive a series of read requests for the highly correlated data elements contained in the first page 202A, the third page 202C, and the fifth page 202E. In response to the series of read requests, the operating system 110 may be configured to read the first page 202A, the third page 202C, and the fifth page 202E, which contain the requested memory cells.

[0030] When implementing a joint decoding decision, the operating system 110 may also be configured to read the threshold voltage levels of neighboring memory cells, some which may be contained in neighboring pages. The second page 202B may neighbor the first page 202A. The fourth page 202D may neighbor the third page 202C. The sixth page 202F may neighbor the fifth page 202E. Thus, the operating system 110 may be configured to also read the second page 202B, the fourth page 202D, and the sixth page 202F. However, the reading of the second page 202B, the fourth page 202D, and the sixth page 202F may not be efficient because these pages are not part of the read request.

[0031] According to various embodiments, the page allocation module 116 may be configured to allocate the same highly correlated data elements to neighboring pages, such as the first page 202A, the second page 202B, and the third page 202C. In this case, the operating system 110 may receive a series of read requests for the highly correlated data elements contained in the first page 202A, the second page 202B, and the third page 202C. In response to the series of read requests, the operating system 110 may read the first page 202A, the second page 202B, and the third page 202C.

[0032] When implementing a joint decoding decision, the operating system 110 may also read the threshold voltages of neighboring memory cells, some which may be contained in other neighboring pages. The second page 202B may neighbor the first page 202A. The

third page 202C may neighbor the second page 202B. The fourth page 202D may neighbor the third page 202C. Here, it can be observed that some of the neighboring pages (e.g., the second page 202B and the third page 202C) overlap with the pages that are read that contain the requested memory cells. Thus, the overlapping page can be read once, and the operating system 110 may be configured to also read the fourth page 202D. By utilizing the page allocation module 116 over conventional page allocation mechanisms, the total number of pages that are read can be reduced from six pages to four pages in this example. Thus, the page allocation module 116 can result in more efficient implementation of a joint decoding decision by the operating system 110.

[0033] FIG. 3 is a flow diagram illustrating an example process 300 adapted to allocate pages in a flash memory, arranged in accordance with at least some embodiments presented herein.. The process 300 may include various operations, functions, or actions as illustrated by one or more blocks 302-308.

[0034] The process 300 may begin at block 302 (Receive Write Request Including Multiple Data Elements), where the page allocation module 116 may be configured to receive a write request that includes multiple data elements, such as the data elements 114, to be written into the flash memory 102. For example, the write request may be received from the application 112. The data elements may include files, file segments, or some other distinguishable units of data. Block 302 may be followed by block 304.

[0035] At block 304 (Determine One or More Groups of Highly Correlated Data Elements), the page allocation module 116 may be configured to identify various correlations between two or more of the multiple data elements. These correlations can be utilized to form groups of highly correlated data elements. Two or more data elements may be highly correlated if the data elements are accessed concurrently, sequentially, or within a threshold amount of time.

[0036] In a first example, a web page may include a HyperText Markup Language (“HTML”) file and an image file. The HTML file and the image file may be considered highly correlated because the files may be displayed together within the same web page. In a second example, a video file may be composed of multiple video frames. Each video frame may correspond to a file segment of the video file. File segments corresponding consecutive video frames may be considered highly correlated because the file segments may be processed in sequence when playing the video file. In a third example, when an e-commerce website displays information regarding a first product as requested by a customer, the e-commerce website may also display information regarding a similar or corresponding second product as a suggestion for the customer. The information regarding the first product and the information regarding the second product may be considered highly correlated because the information for both products may be concurrent displayed via the e-commerce website.

[0037] In some embodiments, data elements may be determined to be highly correlated by monitoring application interaction with the data elements. For example, the application 112 may access (e.g., read and/or write) certain data elements concurrently, sequentially, or within a threshold amount of time. A monitoring module 124 may be configured to monitor and record the time at which each data element is accessed by the application 112. The access time of each data element may be stored in the monitoring record 118. When the operating system 110 receives a write request to store multiple data elements to the flash memory 102, the page allocation module 116 may be configured to identify highly correlated groups of data elements by analyzing the access times of the data elements in the monitoring record 118.

[0038] In some other embodiments, data elements may be determined to be highly correlated based on the heuristics 120. A heuristic may refer to a predefined rule that can be applied to identify certain data elements as being highly correlated. The data elements may be correlated or distinguished based on an identifier (e.g., a file name extension, embedded metadata, external metadata, file header, etc.) associated with or contained within each of the data elements. In one example, a data structure may be embodied in data elements that span multiple pages. In such a case, the heuristics 120 may specify that data elements that are part of the data structure are highly correlated. In another example, an audio file may be embodied in multiple data elements that span multiple pages. In such a case, the heuristics 120 may specify that data elements representing consecutive audio segments in the audio file are highly correlated.

[0039] In yet some other embodiments, data elements may be determined to be highly correlated based on user interaction with an application. For example, the application 112 may enable a user to flag or highlight certain data elements as being highly correlated. The user-provided correlations may be stored in the user-interaction record 122. When the operating system 110 receives a write request to store multiple data elements to the flash memory 102, the page allocation module 116 may be configured to identify highly correlated groups of data elements by utilizing the correlations stored in the user-interaction record 122.

[0040] The page allocation module 116 may be configured to determine the highly correlated groups using the monitoring record 118, the heuristics 120, and/or the user-interaction record 122. The page allocation module 116 may also be configured to compute a correlation score for each group of data elements. A higher correlation score may represent a higher correlation, while a lower correlation score may represent a lesser correlation between two or more data elements. Data elements may be considered highly correlated if they meet a minimum correlation score. The correlation score may be determined based on a statistical model associated with the monitoring record, the heuristics 120, and/or the user-interaction record 122. The statistical model may specify that some properties or combinations of

properties may be associated with a higher correlation score, while some other properties or other combinations of properties may be associated with a lower correlation score. Block 304 may be followed by block 306.

[0041] At block 306, (Allocate Neighboring Pages in Flash Memory to Each Group of Highly Correlated Data Elements), the page allocation module 116 may be configured to allocate neighboring pages in the flash memory 102, as possible, to each group of highly correlated data elements. In some embodiments, neighboring pages may be adjacent pages within the flash memory 102. For example, the adjacent pages may have consecutive memory addresses. In some other embodiments, neighboring pages may be near-adjacent pages within the flash memory 102. In some case, free adjacent pages may not be available, so near-adjacent pages that are physically closer may be allocated. Block 306 may be followed by block 308.

[0042] At block 308 (Write Each Group of Highly Correlated Data Elements to Allocated Neighboring Pages), the page allocation module 116 may be configured to write each group of highly correlated data elements to the allocated neighboring pages in the flash memory 102. After block 308, the process 300 either repeat (e.g., periodically, continuously, or on demand as needed) or terminate.

[0043] FIG. 4 is a block diagram illustrating a computer hardware architecture for an example computing system, arranged in accordance with at least some embodiments presented herein. FIG. 4 includes a computer 400, including a processor 410, memory 420, and one or more drives 430. The computer 400 may be implemented as a conventional computer system, an embedded control computer, a laptop, or a server computer, a mobile device, a set-top box, a kiosk, a vehicular information system, a mobile telephone, a customized machine, or other hardware platform.

[0044] The drives 430 and their associated computer storage media, provide storage of computer readable instructions, data structures, program modules and other data for the computer 400. The drives 430 can include an operating system 440, application programs 450, program modules 460, and a database 480. The operating system 540 may be configured to include the page allocation module 116 for allocating pages in the flash memory 102. The computer 400 further includes user input devices 490 through which a user may enter commands and data. Input devices can include an electronic digitizer, a microphone, a keyboard and pointing device, commonly referred to as a mouse, trackball or touch pad. Other input devices may include a joystick, game pad, satellite dish, scanner, or the like.

[0045] These and other input devices can be coupled to the processor 410 through a user input interface that is coupled to a system bus, but may be coupled by other interface and bus structures, such as a parallel port, game port or a universal serial bus (“USB”).

Computers such as the computer 400 may also include other peripheral output devices such as speakers, which may be coupled through an output peripheral interface 494 or the like.

[0046] The computer 400 may operate in a networked environment using logical connections to one or more computers, such as a remote computer coupled to a network interface 496. The remote computer may be a personal computer, a server, a router, a network PC, a peer device or other common network node, and can include many or all of the elements described above relative to the computer 400. Networking environments are commonplace in offices, enterprise-wide area networks (“WAN”), local area networks (“LAN”), intranets, and the Internet.

[0047] When used in a LAN or WLAN networking environment, the computer 400 may be coupled to the LAN through the network interface 496 or an adapter. When used in a WAN networking environment, the computer 400 typically includes a modem or other means for establishing communications over the WAN, such as the Internet or the network 406. The WAN may include the Internet, the illustrated network 406, various other networks, or any combination thereof. It will be appreciated that other mechanisms of establishing a communications link, ring, mesh, bus, cloud, or network between the computers may be used.

[0048] According to some embodiments, the computer 400 may be coupled to a networking environment. The computer 400 may include one or more instances of a physical computer-readable storage medium or media associated with the drives 430 or other storage devices. The system bus may enable the processor 410 to read code and/or data to/from the computer-readable storage media. The media may represent an apparatus in the form of storage elements that are implemented using any suitable technology, including but not limited to semiconductors, magnetic materials, optical media, electrical storage, electrochemical storage, or any other such storage technology. The media may represent components associated with memory 420, whether characterized as RAM, ROM, flash, or other types of volatile or nonvolatile memory technology. The media may also represent secondary storage, whether implemented as the storage drives 430 or otherwise. Hard drive implementations may be characterized as solid state, or may include rotating media storing magnetically-encoded information.

[0049] The storage media may include one or more program modules 460. The program modules 460 may include software instructions that, when loaded into the processor 410 and executed, transform a general-purpose computing system into a special-purpose computing system. As detailed throughout this description, the program modules 460 may provide various tools or techniques by which the computer 400 may participate within the overall systems or operating environments using the components, logic flows, and/or data structures discussed herein.

[0050] The processor 410 may be constructed from any number of transistors or other circuit elements, which may individually or collectively assume any number of states. More specifically, the processor 410 may operate as a state machine or finite-state machine. Such a machine may be transformed to a second machine, or specific machine by loading executable instructions contained within the program modules 460. These computer-executable instructions may transform the processor 410 by specifying how the processor 410 transitions between states, thereby transforming the transistors or other circuit elements constituting the processor 410 from a first machine to a second machine. The states of either machine may also be transformed by receiving input from the one or more user input devices 490, the network interface 496, other peripherals, other interfaces, or one or more users or other actors. Either machine may also transform states, or various physical characteristics of various output devices such as printers, speakers, video displays, or otherwise.

[0051] Encoding the program modules 460 may also transform the physical structure of the storage media. The specific transformation of physical structure may depend on various factors, in different implementations of this description. Examples of such factors may include, but are not limited to: the technology used to implement the storage media, whether the storage media are characterized as primary or secondary storage, and the like. For example, if the storage media are implemented as semiconductor-based memory, the program modules 460 may transform the physical state of the semiconductor memory 420 when the software is encoded therein. For example, the software may transform the state of transistors, capacitors, or other discrete circuit elements constituting the semiconductor memory 420.

[0052] As another example, the storage media may be implemented using magnetic or optical technology such as drives 430. In such implementations, the program modules 460 may transform the physical state of magnetic or optical media, when the software is encoded therein. These transformations may include altering the magnetic characteristics of particular locations within given magnetic media. These transformations may also include altering the physical features or characteristics of particular locations within given optical media, to change the optical characteristics of those locations. It should be appreciated that various other transformations of physical media are possible without departing from the scope and spirit of the present description.

[0053] FIG. 5 is a schematic diagram that illustrates a computer program product 500 that includes a computer program for executing a computer process on a computing device, arranged in accordance with at least some embodiments presented herein. An illustrative embodiment of the example computer program product is provided using a signal bearing medium 502, and may include at least one instruction of 504: one or more instructions for receiving a plurality of data elements and a write request to write the plurality

of data elements to a flash memory; one or more instructions for identifying a correlation between a subset of the data elements based on correlation criteria; one or more instructions for allocating neighboring pages of the flash memory for storing the subset of the data elements; or one or more instructions for writing the subset of the data elements into the allocated pages. In some embodiments, the signal bearing medium 502 of the one or more computer program products 500 include a computer readable medium 506, a recordable medium 508, and/or a communications medium 510.

[0054] While the subject matter described herein is presented in the general context of program modules that execute in conjunction with the execution of an operating system and application programs on a computer system, those skilled in the art will recognize that other implementations may be performed in combination with other types of program modules. Generally, program modules include routines, programs, components, data structures, and other types of structures that perform particular tasks or implement particular abstract data types. Moreover, those skilled in the art will appreciate that the subject matter described herein may be practiced with other computer system configurations, including hand-held devices, multi-core processor systems, microprocessor-based or programmable consumer electronics, minicomputers, mainframe computers, and the like.

[0055] The present disclosure is not to be limited in terms of the particular embodiments described in this application, which are intended as illustrations of various aspects. Many modifications and variations can be made without departing from its spirit and scope, as will be apparent to those skilled in the art. Functionally equivalent methods and apparatuses within the scope of the disclosure, in addition to those enumerated herein, will be apparent to those skilled in the art from the foregoing descriptions. Such modifications and variations are intended to fall within the scope of the appended claims. The present disclosure is to be limited only by the terms of the appended claims, along with the full scope of equivalents to which such claims are entitled. It is to be understood that this disclosure is not limited to particular methods, reagents, compounds compositions or biological systems, which can, of course, vary. It is also to be understood that the terminology used herein is for the purpose of describing particular embodiments only, and is not intended to be limiting.

[0056] With respect to the use of substantially any plural and/or singular terms herein, those having skill in the art can translate from the plural to the singular and/or from the singular to the plural as is appropriate to the context and/or application. The various singular/plural permutations may be expressly set forth herein for sake of clarity.

[0057] It will be understood by those within the art that, in general, terms used herein, and especially in the appended claims (*e.g.*, bodies of the appended claims) are generally intended as “open” terms (*e.g.*, the term “including” should be interpreted as “including but not limited to,” the term “having” should be interpreted as “having at least,”

the term “includes” should be interpreted as “includes but is not limited to,” etc.). It will be further understood by those within the art that if a specific number of an introduced claim recitation is intended, such an intent will be explicitly recited in the claim, and in the absence of such recitation no such intent is present. For example, as an aid to understanding, the following appended claims may contain usage of the introductory phrases “at least one” and “one or more” to introduce claim recitations. However, the use of such phrases should not be construed to imply that the introduction of a claim recitation by the indefinite articles “a” or “an” limits any particular claim containing such introduced claim recitation to embodiments containing only one such recitation, even when the same claim includes the introductory phrases “one or more” or “at least one” and indefinite articles such as “a” or “an” (*e.g.*, “a” and/or “an” should be interpreted to mean “at least one” or “one or more”); the same holds true for the use of definite articles used to introduce claim recitations. In addition, even if a specific number of an introduced claim recitation is explicitly recited, those skilled in the art will recognize that such recitation should be interpreted to mean at least the recited number (*e.g.*, the bare recitation of “two recitations,” without other modifiers, means at least two recitations, or two or more recitations). Furthermore, in those instances where a convention analogous to “at least one of A, B, and C, etc.” is used, in general such a construction is intended in the sense one having skill in the art would understand the convention (*e.g.*, “a system having at least one of A, B, and C” would include but not be limited to systems that have A alone, B alone, C alone, A and B together, A and C together, B and C together, and/or A, B, and C together, etc.). In those instances where a convention analogous to “at least one of A, B, or C, etc.” is used, in general such a construction is intended in the sense one having skill in the art would understand the convention (*e.g.*, “a system having at least one of A, B, or C” would include but not be limited to systems that have A alone, B alone, C alone, A and B together, A and C together, B and C together, and/or A, B, and C together, etc.). It will be further understood by those within the art that virtually any disjunctive word and/or phrase presenting two or more alternative terms, whether in the Detailed Description, claims, or Figures, should be understood to contemplate the possibilities of including one of the terms, either of the terms, or both terms. For example, the phrase “A or B” will be understood to include the possibilities of “A” or “B” or “A and B.”

[0058] In addition, where features or aspects of the disclosure are described in terms of Markush groups, those skilled in the art will recognize that the disclosure is also thereby described in terms of any individual member or subgroup of members of the Markush group.

[0059] As will be understood by one skilled in the art, for any and all purposes, such as in terms of providing a written description, all ranges disclosed herein also encompass any and all possible subranges and combinations of subranges thereof. Any listed range can be easily recognized as sufficiently describing and enabling the same range being broken down

into at least equal halves, thirds, quarters, fifths, tenths, etc. As a non-limiting example, each range discussed herein can be readily broken down into a lower third, middle third and upper third, etc. As will also be understood by one skilled in the art all language such as “up to,” “at least,” “greater than,” “less than,” and the like include the number recited and refer to ranges which can be subsequently broken down into subranges as discussed above. Finally, as will be understood by one skilled in the art, a range includes each individual member. Thus, for example, a group having 1-3 elements refers to groups having 1, 2, or 3 elements. Similarly, a group having 1-5 elements refers to groups having 1, 2, 3, 4, or 5 elements, and so forth.

[0060] While various aspects and embodiments have been disclosed herein, other aspects and embodiments will be apparent to those skilled in the art. The various aspects and embodiments disclosed herein are for purposes of illustration and are not intended to be limiting, with the true scope and spirit being indicated by the following claims.

What is claimed is:

1. A method for allocating pages in a flash memory, the method comprising:
receiving a plurality of data elements and a write request to write the plurality of data elements to the flash memory;
identifying a correlation between a subset of the data elements based on correlation criteria;
allocating neighboring pages of the flash memory for storing the subset of the data elements; and
writing the subset of the data elements into the allocated pages.
2. The method of claim 1, further comprising:
receiving a series of read requests to read the subset of the data elements; and
reading the subset of the data elements from the allocated pages and additional pages neighboring the allocated pages according to a joint decoding decision that compensates for inter-cell interference.
3. The method of claim 1, wherein the plurality of data elements comprises a plurality of files or file segments.
4. The method of claim 1, wherein identifying the correlation between the subset of the data elements based on the correlation criteria comprises determining that the subset of the data elements are correlated when the subset of the data elements are accessed concurrently, sequentially, or within a threshold amount of time.
5. The method of claim 4, wherein determining that the subset of the data elements are correlated when the subset of the data elements are accessed concurrently, sequentially, or within the threshold amount of time comprises determining that the subset of the data elements are correlated based on a monitoring record of previous accesses of the subset of the data elements.
6. The method of claim 4, wherein determining that the subset of the data elements are correlated when the subset of the data elements are accessed concurrently, sequentially, or within the threshold amount of time comprises determining that the subset of the data elements are correlated based on a heuristic.

7. The method of claim 4, wherein determining that the subset of the data elements are correlated when the subset of the data elements are accessed concurrently, sequentially, or within the threshold amount of time comprises determining that the subset of the data elements are correlated based on a user-interaction record containing user input of the correlation.

8. The method of claim 1, wherein the neighboring pages comprises adjacent pages within a wordline.

9. The method of claim 1, wherein the neighboring pages comprises adjacent pages between multiple wordlines.

10. The method of claim 1, wherein the flash memory comprises a NAND type flash memory.

11. The method of claim 1, wherein the flash memory is a single-level type flash memory.

12. A computer-readable medium comprising computer-executable instructions stored thereon which, when executed by a computer, cause the computer to:

receive a plurality of data elements and a write request to write the plurality of data elements to a flash memory;

identify a correlation between a subset of the data elements based on correlation criteria;

allocate neighboring pages of the flash memory for storing the subset of the data elements; and

write the subset of the data elements into the allocated pages.

13. The computer-readable medium of claim 12, further comprising computer-executable instructions stored thereon which, when executed by the computer, cause the computer to:

receive a series of read requests to read the subset of the data elements; and

read the subset of the data elements from the allocated pages and additional pages neighboring the allocated pages according to a joint decoding decision that compensates for inter-cell interference.

14. The computer-readable medium of claim 12, wherein the plurality of data elements comprises a plurality of files or file segments.

15. The computer-readable medium of claim 12, wherein to identify the correlation between the subset of the data elements based on the correlation criteria, the computer-executable instructions further cause the computer to determine that the subset of the data elements are correlated when the subset of the data elements are accessed concurrently, sequentially, or within a threshold amount of time.

16. The computer-readable medium of claim 12, wherein the subset of the data elements comprises a HyperText Markup Language (HTML) file and an image file that belong to a single web page.

17. The computer-readable medium of claim 12, wherein the subset of the data elements comprises consecutive video frames of a video.

18. The computer-readable medium of claim 12, wherein the subset of the data elements comprises a first data element including information of a first product and a second data element including information of a second product that is related to the first product.

19. The computer-readable medium of claim 12, wherein the neighboring pages comprises adjacent pages within a wordline.

20. A computing device, comprising:
a flash memory comprising a plurality of memory cells arranged to form a plurality of memory pages; and
an operating system coupled to the flash memory and configured to
receive a plurality of data elements and a write request to write the plurality of data elements to the flash memory,
identify a correlation between a subset of the data elements based on correlation criteria,
allocate neighboring pages of the flash memory for storing the subset of the data elements, and
write the subset of the data elements into the allocated pages.

21. The computing device of claim 20, wherein the operating system is further configured to:

receive a series of read requests to read the subset of the data elements; and

read the subset of the data elements from the allocated pages and additional pages neighboring the allocated pages according to a joint decoding decision that compensates for inter-cell interference.

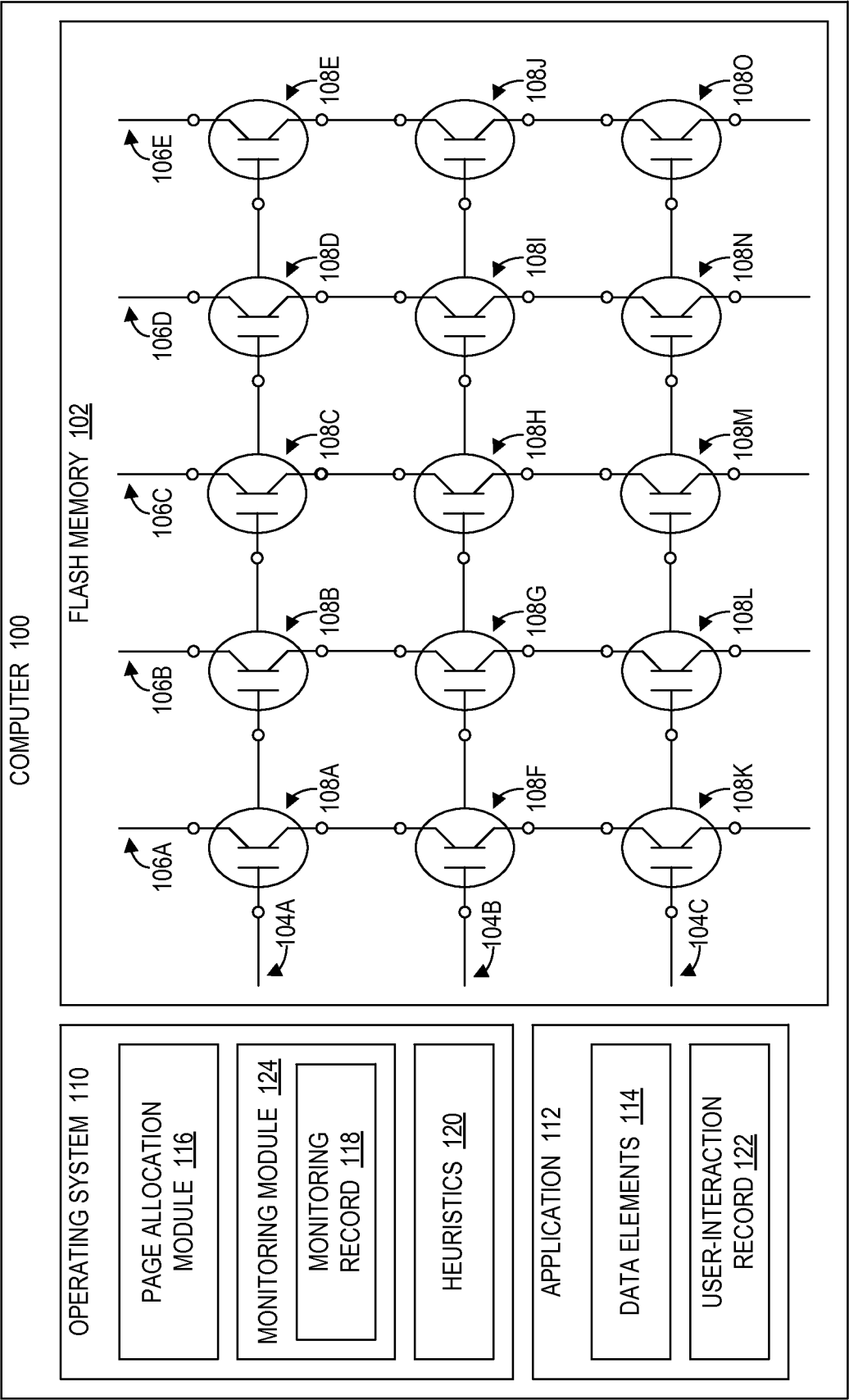


Fig. 1

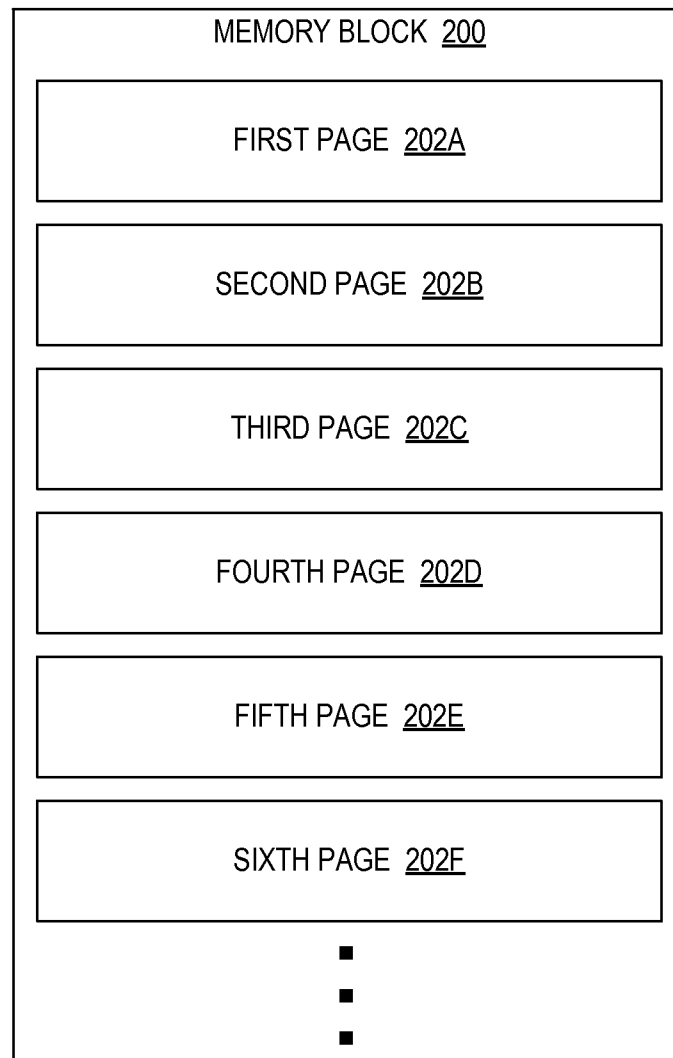
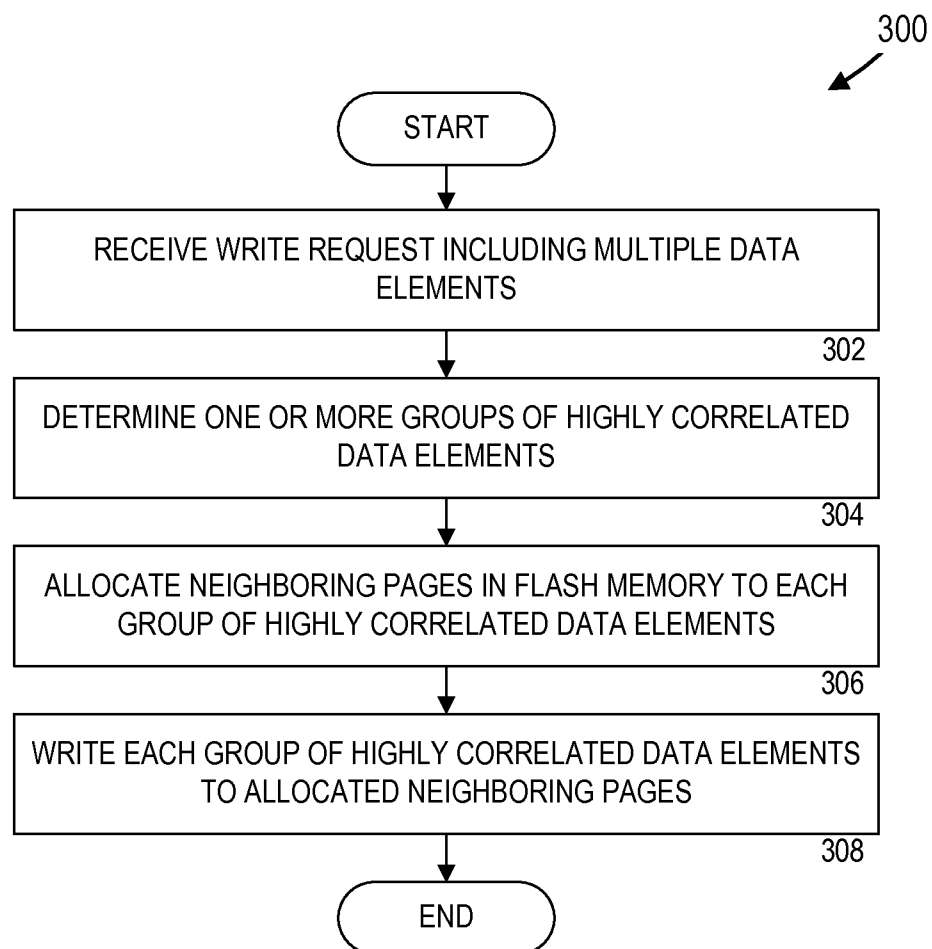


Fig. 2

*Fig. 3*

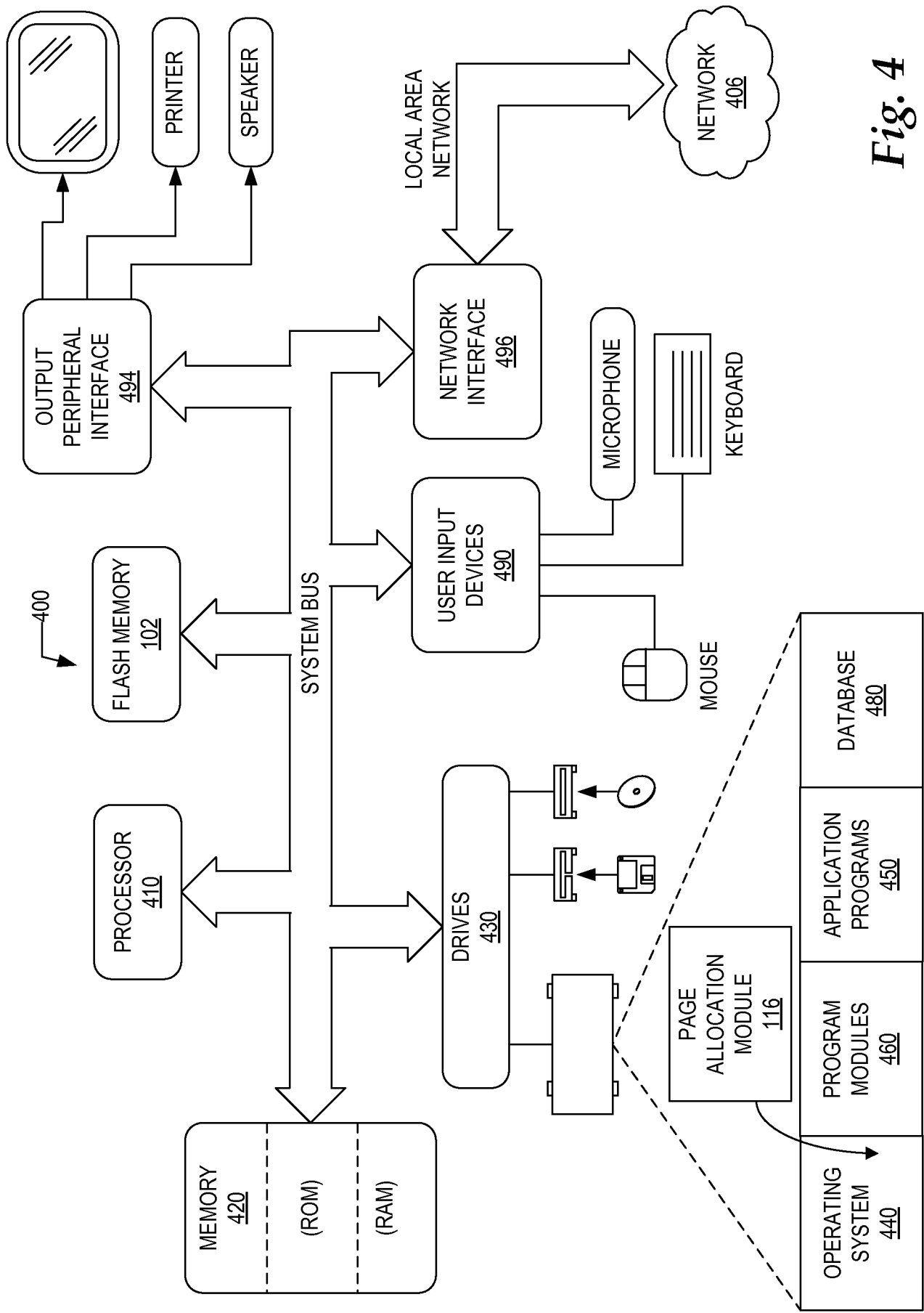


Fig. 4

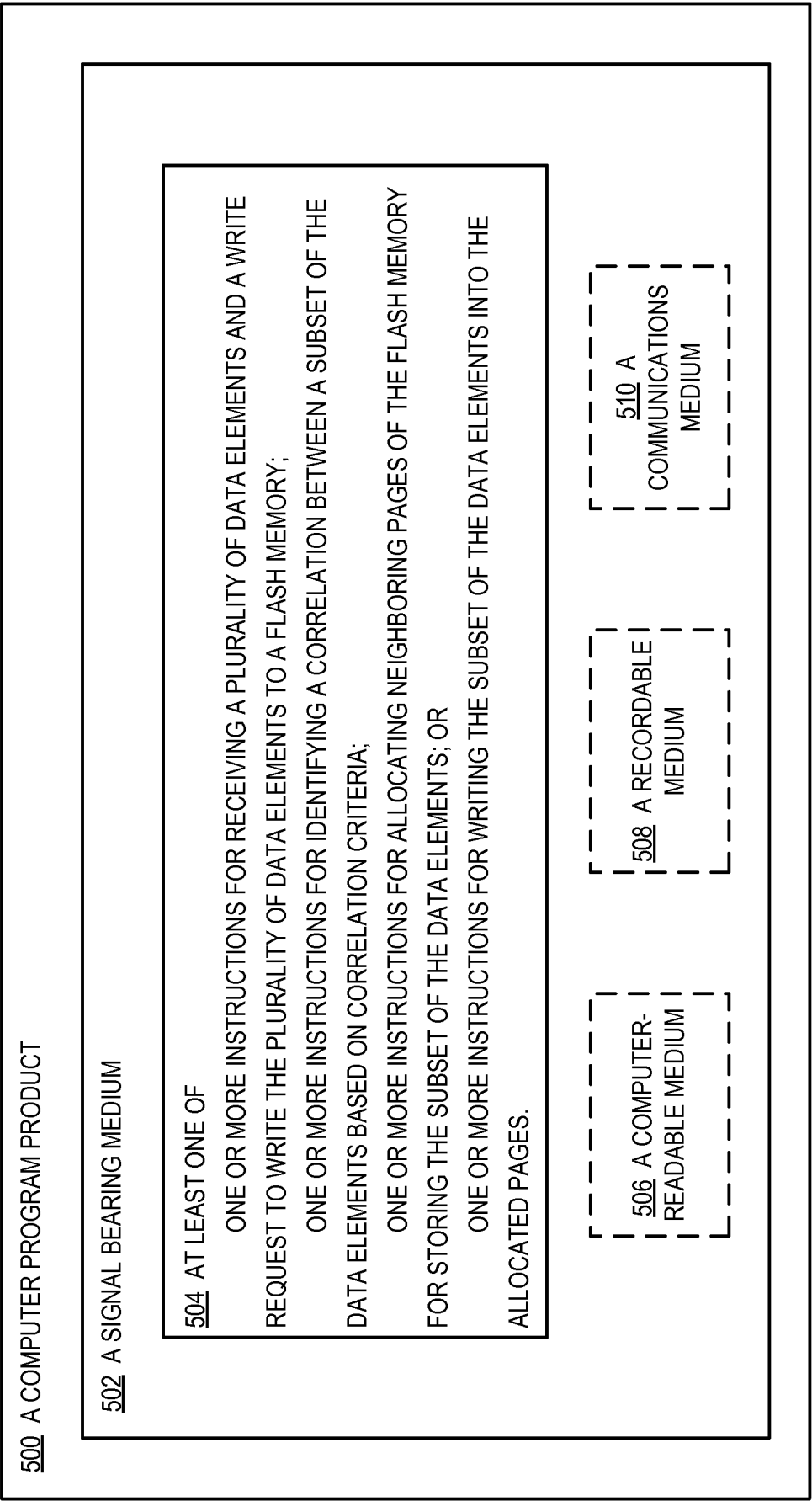


Fig. 5

INTERNATIONAL SEARCH REPORT

International application No.

PCT/US2013/021168

A. CLASSIFICATION OF SUBJECT MATTER

G06F 12/10 (2006.01) G11C 7/00 (2006.01)

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)

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 practicable, search terms used)

EPODOC, WPI: Keywords (flash, usb, nand, relate, data, set, allocate, write, page, block, wordline) and like terms.

Google Patents, Google: Similar keywords as above.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
	Documents are listed in the continuation of Box C	



Further documents are listed in the continuation of Box C



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	"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
"E" earlier application or patent but published on or after the international filing date	"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
"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)	"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
"O" document referring to an oral disclosure, use, exhibition or other means	"&" document member of the same patent family
"P" document published prior to the international filing date but later than the priority date claimed	

Date of the actual completion of the international search
4 March 2013Date of mailing of the international search report
04 March 2013

Name and mailing address of the ISA/AU

AUSTRALIAN PATENT OFFICE
PO BOX 200, WODEN ACT 2606, AUSTRALIA
Email address: pct@ipaustalia.gov.au
Facsimile No.: +61 2 6283 7999

Authorised officer

MD Reza-E Rabbi
AUSTRALIAN PATENT OFFICE
(ISO 9001 Quality Certified Service)
Telephone No. 0262833141

INTERNATIONAL SEARCH REPORT Information on patent family members		International application No. PCT/US2013/021168	
This Annex lists known patent family members relating to the patent documents cited in the above-mentioned international search report. The Australian Patent Office is in no way liable for these particulars which are merely given for the purpose of information.			
Patent Document/s Cited in Search Report		Patent Family Member/s	
Publication Number	Publication Date	Publication Number	Publication Date
US 2011/0107056 A1	05 May 2011	TW 201115582 A	01 May 2011
		US 2011107056 A1	05 May 2011
End of Annex			
Due to data integration issues this family listing may not include 10 digit Australian applications filed since May 2001. Form PCT/ISA/210 (Family Annex)(July 2009)			