



(51) International Patent Classification:

G11C 13/00 (2006.01) G11C 11/56 (2006.01)
G11C 16/34 (2006.01)

(21) International Application Number:

PCT/US2015/030847

(22) International Filing Date:

14 May 2015 (14.05.2015)

(25) Filing Language:

English

(26) Publication Language:

English

(30) Priority Data:

62/005,908 30 May 2014 (30.05.2014) US
14/321,701 1 July 2014 (01.07.2014) US

(71) Applicant: SANDISK TECHNOLOGIES INC.
[US/US]; Two Legacy Town Center, 6900 North Dallas
Parkway, Plano, TX 75024 (US).(72) Inventors: TRUONG, Linh, Tien; 951 SanDisk Drive,
Milpitas, CA 95035 (US). SAMUELS, Allen; 951
SanDisk Drive, Milpitas, CA 95035 (US). KANKANI,
Navneeth; 951 SanDisk Drive, Milpitas, CA 95035 (US).(74) Agents: WILLIAMS, Gary, S. et al.; Morgan Lewis &
Bockius LLP, 2 Palo Alto Square, 3000 El Camino Real,
Suite 700, Palo Alto, CA 94306 (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, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, 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, SA, 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, ST, 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, KM, ML, MR, NE, SN, TD, TG).

Published:

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

(54) Title: METHOD AND SYSTEM FOR RECHARACTERIZING THE STORAGE DENSITY OF A MEMORY DEVICE OR A PORTION THEREOF

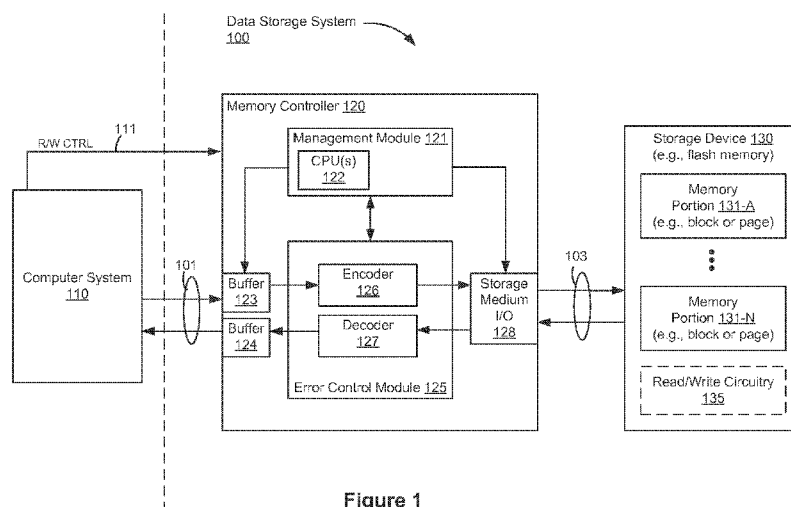


Figure 1

(57) Abstract: A storage system includes a memory controller and a storage device with one or more memory devices, each with a plurality of memory portions. The memory controller determines an initial storage capacity for each of the one or more memory devices, where the one or more memory devices are configured in a first storage density. The memory controller detects a trigger condition as to at least one memory portion of a respective device of the one or more memory devices and, in response to detecting the trigger condition, recharacterizes the at least one memory portion of the respective memory device so as to be configured in a second storage density, where the at least one recharacterized memory portion of the respective memory device has a reduced storage capacity. After the recharacterizing, the memory controller determines a revised storage capacity for the respective memory device.

Method and System for Recharacterizing the Storage Density of a Memory Device or a Portion Thereof

TECHNICAL FIELD

[0001] The disclosed embodiments relate generally to memory systems, and in particular, to improving the reliability with which a storage device (e.g., a flash memory device) retains data.

BACKGROUND

[0002] Non-volatile data storage devices have enabled increased portability of data and software applications. For example, storage elements of a flash memory device may each store multiple bits of data, enhancing data storage density as compared to single-level cell (SLC) flash memory devices. Consequently, flash memory devices enable users to store and access a large amount of data. However, as a number of bits stored per cell increases, bit errors in stored data typically increase, and in addition, MLC memory cells typically wear faster than SLC memory cells. Furthermore, when a flash memory device reaches end-of-life conditions, the memory device is typically retired or removed from a storage system.

SUMMARY

[0003] In some embodiments, a memory controller is configured to perform operations with/on a storage device (e.g., a flash memory device). In some embodiments, the memory controller recharacterizes at least a portion of a memory device from a first storage density to a second storage density after detecting a trigger condition so as to maximize the life of the portion of the memory device, and the memory device itself, past end-of-life conditions for the first storage density.

BRIEF DESCRIPTION OF THE DRAWINGS

[0004] So that the present disclosure can be understood in greater detail, a more particular description may be had by reference to the features of various implementations, some of which are illustrated in the appended drawings. The appended drawings, however, merely illustrate the more pertinent features of the present disclosure and are therefore not to be considered limiting, for the description may admit to other effective features.

[0005] Figure 1 is a block diagram of an implementation of a data storage system in accordance with some embodiments.

[0006] Figure 2A is a block diagram of an implementation of a management module in accordance with some embodiments.

[0007] Figure 2B is a diagram of a characterization vector table included in Figure 2A in accordance with some embodiments.

[0008] Figure 2C is a diagram of a representative characterization vector included in Figure 2B in accordance with some embodiments.

[0009] Figure 3 is a diagram of a first implementation of a recharacterization process in accordance with some embodiments.

[0010] Figure 4 is a diagram of a second implementation of a recharacterization process in accordance with some embodiments.

[0011] Figures 5A-5C illustrate a flowchart representation of a method of operation in a storage system in accordance with some embodiments.

[0012] In accordance with common practice the various features illustrated in the drawings may not be drawn to scale. Accordingly, the dimensions of the various features may be arbitrarily expanded or reduced for clarity. In addition, some of the drawings may not depict all of the components of a given system, method or device. Finally, like reference numerals may be used to denote like features throughout the specification and figures.

DETAILED DESCRIPTION

[0013] Sometimes herein, a memory cell configured to store 1 bit is referred to as an X1 memory cell, a memory cell configured to store 2 bits is referred to as a X2 memory cell, and a memory cell configured to store 3 bits is referred to as an X3 memory cell.

Additionally, a retired memory cell, which is not available for further programming, is referred to as an X0 memory cell. Furthermore, sometimes X3, X2, and X1 are used to indicate levels of storage density: X3 (three bits per memory cell), X2 (two bits per memory cell) and X1 (one bit per memory cell).

[0014] The various implementations described herein include systems, devices, and/or methods that may improve the reliability with which data can be retained by a storage device (e.g., a flash memory device). Some implementations include systems, devices, and/or methods to recharacterization at least a portion of a memory device from a first storage density to a second storage density (e.g., from X2 to X1) so as to maximize the life of the portion of the memory device and the memory device itself. In some implementations, the

memory cells of a memory device, or a portion of the memory device, are recharacterized from X2 to X1 after reaching end-of-life conditions while the memory cells are operating as X2 memory cells, so as to extend the operating life of the memory device.

[0015] In some embodiments, a method of operation in a storage system that includes a memory controller and a storage device with one or more memory devices is performed by the memory controller (e.g., embedded in the storage device or separate from the storage device). In some embodiments, a respective memory device of the one or more memory devices includes a plurality of memory portions (e.g., die, block, or pages portions). The method includes determining an initial storage capacity for each of the one or more memory devices, where the one or more memory devices are configured in a first storage density. The method includes: detecting a trigger condition as to at least one memory portion of a respective device of the one or more memory devices; and, in response to detecting the trigger condition, recharacterizing the at least one memory portion of the respective memory device so as to be configured in a second storage density, where the at least one recharacterized memory portion of the respective memory device has a reduced storage capacity. After the recharacterizing, the method includes determining a revised storage capacity for the respective memory device.

[0016] Some embodiments include an electronic system or device (e.g., data storage system 100, Figure 1 or memory controller 120, Figure 1), comprising: one or more processors; and memory storing one or more programs to be executed by the one or more processors, the one or more programs comprising instructions for performing or controlling performance of any of the methods described herein. Some embodiments include a non-transitory computer readable storage medium, storing one or more programs for execution by one or more processors of an electronic system or device (e.g., data storage system 100, Figure 1 or memory controller 120, Figure 1), the one or more programs including instructions for performing or controlling performance of any of the methods described herein. Some embodiments include an electronic system or device (e.g., data storage system 100, Figure 1 or memory controller 120, Figure 1) comprising means for performing or controlling performance of the operations of any of the methods described herein.

[0017] Numerous details are described herein in order to provide a thorough understanding of the example implementations illustrated in the accompanying drawings. However, some embodiments may be practiced without many of the specific details, and the scope of the claims is only limited by those features and aspects specifically recited in the

claims. Furthermore, well-known methods, components, and circuits have not been described in exhaustive detail so as not to unnecessarily obscure more pertinent aspects of the implementations described herein.

[0018] Figure 1 is a diagram of an implementation of a data storage system 100 in accordance with some embodiments. While some example features are illustrated, various other features have not been illustrated for the sake of brevity and so as not to obscure more pertinent aspects of the example embodiments disclosed herein. To that end, as a non-limiting example, data storage system 100 is used in conjunction with a computer system 110, where data storage system 100 includes a memory controller 120 and a storage device 130. In some embodiments, storage device 130 includes a single memory device (e.g., a volatile memory device or a non-volatile memory (NVM) device such as a magnetic disk storage device, an optical disk storage device, a flash memory device, a three-dimensional (3D) memory device (as further described herein), or another semiconductor NVM memory device). In some embodiments, storage device 130 includes a plurality of memory devices or is one of a plurality of memory devices coupled with memory controller 120. In some embodiments, a memory device includes one or more die, each with two or more individually addressable blocks (e.g., erase blocks). In some embodiments, storage device 130 includes NAND-type flash memory or NOR-type flash memory. Further, in some embodiments, memory controller 120 is a solid-state drive (SSD) controller. However, one or more other types of storage media may be included in accordance with aspects of a wide variety of embodiments.

[0019] Computer system 110 is coupled with memory controller 120 through data connections 101. However, in some embodiments, computer system 110 includes memory controller 120 as a component and/or a sub-system. Computer system 110 may be any suitable computing device, such as a desktop computer, a laptop computer, a tablet device, a netbook, an internet kiosk, a personal digital assistant, a mobile phone, a smart phone, a gaming device, a wearable computing device, a computer server, or any other computing device. Computer system 110 is sometimes called a host or host system. In some embodiments, computer system 110 includes one or more processors, one or more types of memory, a display, and/or other user interface components such as a keyboard, a touch screen display, a mouse, a track-pad, a digital camera, and/or any number of supplemental devices to add I/O functionality.

[0020] Storage device 130 is coupled with memory controller 120 through connections 103. Connections 103 are sometimes called data connections, but typically

convey commands in addition to data, and optionally convey metadata, error correction information, and/or other information in addition to data values to be stored in storage device 130 and data values read from storage device 130. In some embodiments, however, memory controller 120 and storage device 130 are included in the same device as components thereof. Furthermore, in some implementations memory controller 120 and storage device 130 are embedded in a host device, such as a mobile device, tablet, other computer, or computer controlled device, and the methods described herein are performed by the embedded memory controller. Storage device 130 may include any number (i.e., one or more) of memory devices including, without limitation, non-volatile semiconductor memory devices, such as flash memory. For example, flash memory devices can be configured for enterprise storage suitable for applications such as cloud computing, or for caching data stored (or to be stored) in secondary storage, such as hard disk drives. Additionally and/or alternatively, flash memory can also be configured for relatively smaller-scale applications such as personal flash drives or hard-disk replacements for personal, laptop and tablet computers.

[0021] In Figure 1, storage device 130 (e.g., with one or more memory devices) includes a plurality of memory portions 131-A, ..., 131-N. For example, a respective memory portion is one of a plurality of memory devices in storage device 130, or a die, block (e.g., an individually addressable block, such as an erase block), word line, or page associated with a respective memory device in storage device 130. In another example, a respective memory portion is a set of memory units (e.g., erase blocks, such as one per die) in multiple die.

[0022] In some embodiments, storage device 130 is divided into a number of individually addressable (and thus individually selectable) blocks. In some embodiments, the individually selectable blocks are the minimum size erasable units in a flash memory device. In other words, each block contains the minimum number of memory cells that can be erased simultaneously. Each block is usually further divided into a plurality of pages and/or word lines, where each page or word line is typically an instance of the smallest individually accessible (readable) portion in a block. In some embodiments (e.g., using some types of flash memory), the smallest individually accessible unit of a data set, however, is a sector, which is a subunit of a page. That is, a block includes a plurality of pages, each page contains a plurality of sectors, and each sector is the minimum unit of data for reading data from the flash memory device. In some embodiments, storage device 130 includes read/write circuitry 135 for selecting a respective portion of storage device 130 on which to perform a memory

operation (e.g., a read, write, or erase operation) and for causing performance of the memory operation on the respective portion of storage device 130.

[0023] For example, one block includes a number of pages (e.g., 64 pages, 128 pages, 256 pages, or another suitable number of pages). In some implementations, the blocks in a die are grouped into a plurality of zones. Typically, each block zone of the die is in a physically distinct region of the die such as a particular half or particular quadrant of the memory cell array in the die. In some implementations, each block zone is independently managed to some extent, which increases the degree of parallelism for parallel operations and simplifies management of storage device 130.

[0024] In some embodiments, memory controller 120 includes a management module 121, an input buffer 123, an output buffer 124, an error control module 125, and a storage medium interface (I/O) 128. In some embodiments, memory controller 120 includes various additional features that have not been illustrated for the sake of brevity and so as not to obscure more pertinent features of the example embodiments disclosed herein, and that a different arrangement of features may be possible. Input buffer 123 and output buffer 124 provide an interface to computer system 110 through data connections 101. Similarly, storage medium I/O 128 provides an interface to storage device 130 through connections 103. In some embodiments, storage medium I/O 128 includes read/write circuitry capable of providing reading signals to storage device 130. For example, the reading signals specify reading threshold voltages for NAND-type flash memory.

[0025] In some embodiments, management module 121 includes one or more processing units (CPU(s), also sometimes called one or more processors) 122 configured to execute instructions in one or more programs (e.g., in management module 121). In some embodiments, one or more CPUs 122 are shared by one or more components within, and in some cases, beyond the function of memory controller 120. Management module 121 is coupled with input buffer 123, output buffer 124 (connection not shown), error control module 125, and storage medium I/O 128 in order to coordinate the operation of these components.

[0026] Error control module 125 is coupled with storage medium I/O 128, input buffer 123 and output buffer 124. Error control module 125 is provided to limit the number of uncorrectable errors inadvertently introduced into data. In some embodiments, error control module 125 is executed in software by one or more CPUs 122 of management module 121,

and, in other embodiments, error control module 125 is implemented in whole or in part using special purpose circuitry to perform encoding and decoding functions associated with error correction. To that end, error control module 125 includes an encoder 126 and a decoder 127. In some embodiments, error control module 125 is configured to encode data (i.e., with encoder 126) and decode raw read data (i.e., with decoder 127) according to one of a plurality of error control code (ECC) techniques, or ECC strengths, such as Reed-Solomon (RS), turbo-code, Bose-Chaudhuri-Hocquenghem (BCH), low-density parity check (LDPC), or other error control codes, or a combination thereof.

[0027] Those skilled in the art will appreciate that various error control codes have different error detection and correction capacities, and that particular codes are selected for various applications for reasons beyond the scope of this disclosure. As such, an exhaustive review of the various types of error control codes is not provided herein. Moreover, those skilled in the art will appreciate that each type or family of error control codes may have encoding and decoding algorithms that are particular to the type or family of error control codes. On the other hand, some algorithms may be utilized at least to some extent in the decoding of a number of different types or families of error control codes. As such, for the sake of brevity, an exhaustive description of the various types of encoding and decoding algorithms generally available and known to those skilled in the art is not provided herein.

[0028] During a write operation, input buffer 123 receives data to be stored in storage device 130 from computer system 110 (e.g., write data). The data held in input buffer 123 is made available to encoder 126, which encodes the data by applying an error control code to produce one or more codewords. The one or more codewords are made available to storage medium I/O 128, which transfers the one or more codewords to storage device 130 in a manner dependent on the type of storage medium being utilized.

[0029] A read operation is typically initiated when computer system (host) 110 sends one or more host read commands on control line 111 to memory controller 120 requesting data from storage device 130. Memory controller 120 sends one or more read access commands to storage device 130, via storage medium I/O 128, to obtain raw read data, typically in accordance with memory locations (addresses) specified by the one or more host read commands. Storage medium I/O 128 provides the raw read data (e.g., comprising one or more codewords) to decoder 127. Decoder 127 applies a decoding process to the encoded data to recover the data, and to correct errors in the recovered data within the error correcting capability of the error control code. If the decoding is successful, the decoded data is

provided to output buffer 124, where the decoded data is made available to computer system 110. Optionally, decoder 127 provides an indication of a number of corrected bits. In some embodiments, if the decoding is not successful, memory controller 120 may resort to a number of remedial actions or provide an indication of an irresolvable error condition.

[0030] Flash memory devices utilize memory cells to store data as electrical values, such as electrical charges or voltages. Each flash memory cell typically includes a single transistor with a floating gate that is used to store a charge, which modifies the threshold voltage of the transistor (i.e., the voltage needed to turn the transistor on). The magnitude of the charge, and the corresponding threshold voltage the charge creates, is used to represent one or more data values. In some implementations, during a read operation, a reading threshold voltage is applied to the control gate of the transistor and the resulting sensed current or voltage is mapped to a data value.

[0031] The terms “cell voltage” and “memory cell voltage,” in the context of flash memory cells, means the threshold voltage of the memory cell, which is the minimum voltage that needs to be applied to the gate of the memory cell’s transistor in order for the transistor to conduct current. Similarly, reading threshold voltages (sometimes also called reading signals and reading voltages) applied to a flash memory cells are gate voltages applied to the gates of the flash memory cells to determine whether the memory cells conduct current at that gate voltage. In some implementations, when a flash memory cell’s transistor conducts current at a given reading threshold voltage, indicating that the cell voltage is less than the reading threshold voltage, the raw data value for that read operation is a “1” and otherwise the raw data value is a “0.”

[0032] Figure 2A is a block diagram illustrating an exemplary management module 121, in accordance with some embodiments. Management module 121 typically includes one or more processing units 122 (also sometimes called CPU(s), processing unit(s), microprocessor(s), microcontroller(s), or core(s)) for executing modules, programs and/or instructions stored in memory 206 and thereby performing processing operations, memory 206, and one or more communication buses 208 for interconnecting these components. Communication buses 208 optionally include circuitry (sometimes called a chipset) that interconnects and controls communications between system components. Management module 121 is coupled with buffer 123, buffer 124, error control module 125, and storage device 130 by communication buses 208. Memory 206 includes volatile memory (e.g., one or more high-speed random access memory devices such as DRAM, SRAM, DDR RAM, or

other random access solid state memory devices), and/or non-volatile memory (e.g., one or more NVM devices such as magnetic disk storage device(s), optical disk storage device(s), flash memory device(s), 3D memory device(s), or other non-volatile solid state storage device(s)). Memory 206 optionally includes one or more storage devices remotely located from one or more processing units 122. Memory 206, or alternately the non-volatile memory device(s) within memory 206, comprises a non-transitory computer readable storage medium. In some embodiments, memory 206, or the computer readable storage medium of memory 206 stores the following programs, modules, and data structures, or a subset or superset thereof:

- data read module 212 for reading data, or causing data to be read, from storage device 130;
- data write module 214 for writing data, or causing data to be written, to storage device 130;
- data erase module 216 for erasing data, or causing data to be erased, from storage device 130;
- request handling module 218 for receiving queries and commands from the host (e.g., computer system 110, Figure 1) and/or queries and commands from other internal processes;
- reporting module 219 for reporting, to the host (e.g., computer system 100, Figure 1), the storage capacity and/or status metric(s) associated with a respective portion of storage device 130;
- wear leveling module 220 for optimally determining pages or blocks of storage device 130 for storing data so as to evenly wear the pages or blocks of storage device 130;
- metric maintaining module 222 for maintaining one or more status metrics for each memory portion of the plurality of memory portions of storage device 130 (e.g., a memory device, die, block, word line, or page portion) or a combined status metric for each memory portion of storage device 130, where the combined status metric for a respective memory portion is computed based on a predefined algorithm that includes the one or more status metrics corresponding to the respective memory portion;
- trigger detection module 224 for detecting a trigger condition as to a respective portion of storage device 130;

- migration module 226 for optionally migrating data stored in the respective portion of storage device 130 (e.g., to another portion storage device 130 such as a different memory device or block associated with data storage system 100)
- recharacterization module 228 for recharacterizing the storage density of the respective portion of storage device 130 in response to detecting the trigger condition as to the respective portion of storage device 130;
- characterization vector table 234 storing a collection of characterization vectors 235 (Figure 2B) that store characterization data for respective portions of storage device 130; and
- logical-to-physical mapping 236 storing a logical-to-physical map which maps logical addresses recognized by the host (e.g., computer system 110, Figure 1) to physical addresses of storage device 130.

[0033] Each of the above identified elements may be stored in one or more of the previously mentioned memory devices, and corresponds to a set of instructions for performing a function described above. The above identified modules or programs (i.e., sets of instructions) need not be implemented as separate software programs, procedures or modules, and thus various subsets of these modules may be combined or otherwise rearranged in various embodiments. In some embodiments, memory 206 may store a subset of the modules and data structures identified above. Furthermore, memory 206 may store additional modules and data structures not described above. In some embodiments, the programs, modules, and data structures stored in memory 206, or the computer readable storage medium of memory 206, provide instructions for implementing any of the methods described below with reference to Figures 5A-5C. In some embodiments, the above identified modules may be implemented in hardware, firmware, software, or a combination thereof. Further, one or more of the above identified modules may be implemented in whole or in part in hardware, for example, using one or more state machines.

[0034] Although Figure 2A shows a management module 121, Figure 2A is intended more as functional description of the various features which may be present in a management module than as a structural schematic of the embodiments described herein. In practice, and as recognized by those of ordinary skill in the art, items shown separately could be combined and some items could be separated.

[0035] Figure 2B is a block diagram illustrating an implementation of a characterization vector table 234 in accordance with some embodiments. Characterization vector table 234 includes a collection of characterization vectors 235, optionally implemented as a collection of tuples for efficient storage, that each store characterization data associated with a respective portion of storage device 130 (e.g., a distinct memory device, die, block zone, block, word line, word line zone, or page portion of storage device 130). In some embodiments, each vector (e.g., vector 235-1, vector 235-2, ..., vector 235-N) in the collection of characterization vectors 235 stores characterization data (e.g., as shown in Figure 2C, bytes written 260, PE cycles 262, bit error rate (BER) 264, and/or other usage information 266) derived during each of at least two time periods for a respective portion of storage device 130. In some implementations, the characterization data stored in characterization vectors 235 is statistically derived.

[0036] For example, without limitation, in some embodiments in which storage device 130 (Figure 1) includes multiple memory devices, characterization vector table 234 includes at least one characterization vector for each distinct memory device of storage device 130. In another example, without limitation, in some embodiments in which storage device 130 (Figure 1) includes multiple blocks, characterization vector table 234 includes at least one characterization vector for each distinct block of storage device 130. In another example, in some embodiments, characterization vector table 234 includes a set of distinct characterization vectors 235 for each block of storage device 130, and the set of distinct characterization vectors 235 for each die includes at least one distinct characterization vector for each word line or page in the block. More detailed example implementations of characterization vectors 235 are described below with reference to Figure 2C.

[0037] Figure 2C is a schematic diagram of an implementation of a representative characterization vector 250 (e.g., corresponding to any one of characterization vectors 235 shown in Figure 2B) in accordance with some embodiments. In some embodiments, for a respective portion of storage device 130, characterization data stored in characterization vector 250 includes: (i) a storage density field 252 indicating the current storage density of memory cells in the respective portion of storage device 130 such as X3 (i.e., 3 bits per memory cell), X2 (i.e., 2 bits per memory cell), X1 (i.e., SLC or 1 bit per memory cell), or X0 (i.e., retired memory cells); (ii) a free space field 254 indicating the amount of free space available in the respective portion of storage device 130; and (iii) a storage capacity field 256

indicating the advertised capacity to the host (e.g., computer system 110, Figure 1) or the total capacity of the respective portion of storage device 130.

[0038] In some embodiments, characterization data stored in characterization vector 250 for the respective portion of storage device 130 also includes one or more status metrics 258, non-limiting example of which include: (a) a bytes written field 260 indicating a number of bytes of data written to the respective portion of storage device 130; (b) a PE cycle field 262 indicating a current count of the number of PE cycles performed on the respective portion of storage device 130; (c) a bit error rate (BER) field 264 indicating a number of errors included in a codeword read from a page of the respective portion of storage device 130 or an average number of errors included in the previous N codewords read from page(s) of the respective portion of storage device 130; and (d) other usage information 266 indicating the health, performance, and/or endurance of the respective portion of storage device 130.

[0039] In some embodiments, characterization data stored in characterization vector 250 optionally includes a combined status metric (represented by status metric 258 in Figure 2C) for the respective portion of storage device 130. In some embodiments, the combined status metric is associated with the output of a predefined algorithm (e.g., computed by metric maintaining module 222, Figure 2A) that takes into account one or more usage parameters associated with the respective portion of storage device 130. For example, the predefined algorithm incorporates one or more of: (a) a number of bytes written to the respective portion of storage device 130; (b) a number of PE cycles performed on the respective portion of storage device 130; (c) a BER for codewords read from the respective portion of storage device 130; and (d) other usage information associated with the respective portion of storage device 130 (e.g., temperature, operating conditions, etc.).

[0040] In some embodiments, characterization data stored in characterization vector 250 includes a physical addresses field 268 for the respective portion of storage device 130 that indicates the physical addresses or set of physical addresses corresponding to the respective portion of storage device 130. In some embodiments, characterization data stored in characterization vector 250 for the respective portion of storage device 130 further includes an ECC technique (e.g., RS, BCH, or LDPC) and/or ECC strength (e.g., a strength parameter that indicates or corresponds to the number of parity bits included in each codeword) for encoding data to be written to the respective portion of storage device 130 and decoding data read from the respective portion of storage device 130. In some embodiments,

characterization data stored in characterization vector 250 for the respective portion of storage device 130 further includes read/write parameters (e.g., a programming voltage, number of programming pulses, duration/width/step of programming pulses, and reading threshold voltages for reading different states of memory cells in the respective portion of storage device 130) indicating parameters for writing data to and reading data from the respective portion of storage device 130.

[0041] As indicated above, X3, X2, and X1 are sometimes used to indicate levels of storage density for memory cells: X3 (three bits per memory cell), X2 (two bits per memory cell), and X1 (one bit per memory cell).

[0042] Figure 3 illustrates a first implementation of a recharacterization process 300 in accordance with some embodiments. In one example, recharacterization process 300 is performed on a respective memory device associated with storage device 130 by memory controller 120. In another example, recharacterization process 300 is performed on a portion of a respective memory device associated with storage device 130 (e.g., a die, a block (sometimes called an erase block), or a group or blocks) by memory controller 120. Recharacterization process 300 will be discussed with reference to a respective memory device for ease of discussion; however, one of skill in the art will appreciate how to apply the concepts discussed to a recharacterization process performed on a portion of the respective memory device such as a die, block or group of blocks of the respective memory device.

[0043] In Figure 3, prior to time 304 (i.e., the time identified by time 304), the respective memory device is configured in a first storage density (e.g., X3 or X2). In Figure 3, at or before time 304, memory controller 120 or a component thereof (e.g., trigger detection module 224, Figure 2A) detects a trigger condition as to the respective memory device. In some embodiments, trigger detection module 224 detects the trigger condition as to the respective memory device when the amount of wear remaining for the respective memory device is less than or equal to a predetermined value. In some embodiments, in response to detecting the trigger condition as to the respective memory device, memory controller 120 or a component thereof (e.g., recharacterization module 228, Figure 2A) recharacterizes the respective memory device so as to be configured in a second storage density (e.g., X2 or X1), whereby the memory cells of the respective memory device are able to store less bits per cell in the second storage density than in the first storage density.

[0044] In Figure 3, graph 310 shows the amount of free space available within the respective memory device (e.g., in terms of bytes) on the y-axis versus time or a number of PE cycles on the x-axis; however, graph 310 is discussed with reference to time on the x-axis for ease of discussion. Graph 310 shows that the amount of free space available within the respective memory device decreases from an initial amount (e.g., number of bytes) 312 to a minimum amount (e.g., number of bytes) 314 at time 302. The amount of free space remains constant (or approximately constant) at (or near) the minimum number of bytes 314 (e.g., greater than zero) until time 304, at or after which the respective memory device is recharacterized. In some embodiments, memory controller 120 maintains more than zero bytes of free space for the respective memory device for a number of reasons, including the need for storage space to rewrite valid data during garbage collection operations, and more generally over provisioning for write amplification in storage device 130. In some embodiments, write amplification is caused by garbage collection, wear leveling, random writes, and/or similar operations requiring over provisioning. After recharacterization of the respective memory device, the amount of free space 316 is less than the initial amount of free space 312.

[0045] In Figure 3, graph 320 shows the amount of wear remaining for the respective memory device on the y-axis versus time or a number of PE cycles on the x-axis; however, graph 320 is discussed with reference to time on the x-axis for ease of discussion. For example, the wear remaining is associated with a percentage or ratio based on a number of PE cycles that the respective memory device is rated for versus a count of PE cycles performed on storage device 130. In another example, the wear remaining is associated with a percentage or ratio based on a number of bytes that the respective memory device is rated to write versus the number of bytes written to storage device 130. In another example, the wear remaining is the result of a predefined algorithm that takes into account a count of PE cycle performed on the respective memory device, a BER for data read from the respective memory device, and a number of bytes written to the respective memory (e.g., the wear remaining is the value of combined status metric field 258 for the respective memory device).

[0046] Graph 320 shows that the amount of wear remaining for the respective memory device starts at amount 322 when the respective memory device is configured in the first storage density and decreases. When the respective memory device is recharacterized so as to be configured in the second storage density at time 304, the amount of wear remaining for the respective memory device jumps to amount 324. The initial amount of wear remaining

322, when the respective memory device is configured in the first storage density, is less than amount 324 after the respective memory device is recharacterized so as to be configured in the second storage density. In one example, consistent with graph 320, a number of blocks of the respective memory device that were initially configured for use as X1 blocks are recharacterized as X3 blocks at time 304, resulting in an increased amount of wear remaining. In a second example, consistent with graph 320, most blocks (e.g., 98% or more) of the respective memory device are initially configured for use as X3 blocks, but have relatively small wear life (e.g., 300 P/E cycles), while after recharacterization at 304, the blocks are recharacterized as X1 blocks with much longer wear life (e.g., 10,000 P/E cycles), resulting in an overall increase in wear remaining for the device despite a decrease in total storage capacity.

[0047] In Figure 3, graph 330 shows the total capacity of the respective memory device that is advertised to the host (e.g., in terms of bytes) on the y-axis versus time or a number of PE cycles on the x-axis; however, graph 330 is discussed with reference to time on the x-axis for ease of discussion. Graph 330 shows that the advertised capacity of the respective memory device starts at a first amount of bytes 332. After recharacterization of the respective memory device at time 304, the advertised capacity of the respective memory device is reduced to a second amount of bytes 334 that is less than the first amount of bytes 332.

[0048] Figure 4 illustrates a second implementation of a recharacterization process 400 in accordance with some embodiments. In one example, recharacterization process 400 is performed on a respective memory device associated with storage device 130 by memory controller 120, whereby one or more blocks are recharacterized at each step within window 408. Although window 408 is shown as encompassing six recharacterization operations, in some implementations window 408 encompasses tens, hundreds or thousands of recharacterization operations, each of which recharacterizes one or more blocks of the storage device. In another example, recharacterization process 400 is performed on a block associated with storage device 130 by memory controller 120, whereby one or more pages of the block are recharacterized at each step within window 408. Recharacterization process 400 will be discussed with reference to a respective memory device with six blocks for ease of discussion; however, one of skill in the art will appreciate how to apply the concepts discussed to a recharacterization process performed on a memory device with a different number of blocks (e.g., tens, or hundreds, or even thousands of blocks), or a different number

of memory portions, where a memory portion is recharacterized at each step within window 408.

[0049] In Figure 4, prior to time 404, the respective memory device is configured in a first storage density (e.g., X3 or X2). In Figure 4, at or before time 404, memory controller 120 or a component thereof (e.g., trigger detection module 224, Figure 2A) detects a trigger condition as to the respective memory device. In some embodiments, trigger detection module 224 detects the trigger condition as to the respective memory device when the amount of wear remaining for the respective memory device is less than or equal to a predetermined value that is greater than or equal to zero. In some embodiments, in response to detecting the trigger condition as to the respective memory device, memory controller 120 or a component thereof (e.g., recharacterization module 228, Figure 2A) recharacterizes a first block of the respective memory device so as to be configured in a second storage density (e.g., X2 or X1), whereby the memory cells of the respective memory device are configured to store fewer bits per cell in the second storage density than in the first storage density. As further discussed below, in conjunction with the recharacterization, data previously stored in the first block (if still valid) is deleted and migrated elsewhere. In Figure 4, in window 408 (i.e., from time 404 to time 406), recharacterization module 228 serially recharacterizes blocks of the respective memory device one block at a time until all the blocks in the respective memory device are recharacterized. In some embodiments, two or more blocks of the respective memory device are recharacterized in parallel.

[0050] In Figure 4, graph 410 shows the amount of free space available within the respective memory device (e.g., in terms of bytes) on the y-axis versus time or a number of PE cycles on the x-axis; however, graph 410 is discussed with reference to time on the x-axis for ease of discussion. Graph 410 shows that the amount of free space available within the respective memory device decreases from an initial number of bytes 414 to a predefined number of bytes 416 at time 402. The amount of free space remains constant (or approximately constant) at the predefined number of bytes 416 (e.g., greater than zero) until time 404 when a first block of the respective memory device is recharacterized. In window 408 (i.e., from time 404 to time 406), the amount of free space decreases to zero and subsequently jumps to the predefined number of bytes 416 as the blocks of the respective memory device are serially recharacterized. After time 406 (e.g., when the last remaining block of the respective memory device is recharacterized), the amount of free space remains at the predefined number of bytes 416. In some alternative embodiments, the amount of free

space does not decrease to zero, and instead decreases to a predefined minimum amount that is greater than zero.

[0051] In Figure 4, graph 420 shows the amount of wear remaining for the respective memory device on the y-axis versus time or a number of PE cycles on the x-axis; however, graph 420 is discussed with reference to time on the x-axis for ease of discussion. For example, the wear remaining is associated with a percentage or ratio based on a number of PE cycles that the respective memory device is rated for versus a count of PE cycles performed on storage device 130. In another example, the wear remaining is associated with a percentage or ratio based on a number of bytes that the respective memory device is rated to write versus the number of bytes written to storage device 130. In another example, the wear remaining is the result of a predefined algorithm that takes into account a count of PE cycle performed on the respective memory device, a BER for data read from the respective memory device, and a number of bytes written to the respective memory (e.g., the wear remaining is the value of combined status metric field 258 for the respective memory device).

[0052] Graph 420 shows that the amount of wear remaining for the respective memory device starts at amount 422 when the respective memory device is configured in the first storage density and decreases towards zero. When the first block of the respective memory device is recharacterized so as to be configured in the second storage density at time 404, the amount of wear remaining for the respective memory device starts to increase. In graph 420, the amount of wear remaining continues to increase during window 408, as additional blocks are recharacterized, until it reaches amount 424 at time 406 (i.e., when the last block or group of blocks of the respective memory device is recharacterized). More generally, the amount of wear remaining after recharacterizing the blocks of the respective memory device in window 408 is greater than the wear remaining prior to recharacterizing the blocks of the respective memory device in window 408.

[0053] In Figure 4, graph 430 shows the total capacity of the respective memory device that is advertised to the host (e.g., in terms of bytes) on the y-axis versus time or a number of PE cycles on the x-axis; however, graph 430 is discussed with reference to time on the x-axis for ease of discussion. Graph 430 shows that the advertised capacity of the respective memory device starts at a first amount of bytes 432. During window 408, the advertised capacity of the respective memory device decreases as the blocks of the respective memory device are serially recharacterized. After recharacterization of the blocks of the

respective memory device at time 406, the advertised capacity of the respective memory device is a second amount of bytes 434 that is less than the first amount of bytes 432.

[0054] Figures 5A-5C illustrate a flowchart representation of a method 500 of operation in a storage system with a memory controller and one or more memory devices in accordance with some embodiments. At least in some implementations, method 500 is performed by the memory controller (e.g., memory controller 120, Figure 1) or one or more components thereof (e.g., management module 121, Figures 1 and 2A). In some embodiments, method 500 is performed by a memory controller (e.g., memory controller 120, Figure 1) distinct from and coupled with the storage device (e.g., storage device 130 which includes the one or more memory devices) by one or more connections (e.g., connections 103, Figure 1). A respective memory device of the one or more memory devices includes a plurality of memory portions such as die, groups of blocks, blocks, word lines, pages, or the like. In some embodiments, method 500 is governed by instructions that are stored in a non-transitory computer readable storage medium and the instructions are executed by one or more processors of a device (e.g., one or more processing units (CPUs) 122 of management module 121, Figures 1 and 2A). Optional operations are indicated by dashed lines (e.g., boxes with dashed-line borders).

[0055] The memory controller determines (502) an initial storage capacity for each of the one or more memory devices, where the one or more memory devices are configured in a first storage density. In some embodiments, the memory cells of the one or more memory devices are configured as N bit memory cells (e.g., X1, X2, or X3). In some embodiments, the memory cells of the one or more memory devices are operable at a first storage density (e.g., X3) and also at one or more other (typically lower) storage densities (e.g., X2 and X1). In some embodiments, memory controller 120 determines (e.g., upon startup or power-on of data storage system 100) an initial storage capacity for each of the one or more memory devices associated with storage device 130 based on the storage densities of the one or more memory devices. In some embodiments, after determining the initial storage capacity for a respective memory device, memory controller 120 or a component thereof (e.g., metric maintaining module 222, Figure 2A) stores the initial storage capacity for the respective memory device in storage capacity field 256 of the characterization vector for the respective memory device, where storage capacity field 256 indicates the current storage capacity of the respective memory device.

[0056] In some embodiments, while the one or more memory devices are configured in the first storage density, memory controller 120 writes data to and reads data from the one or more memory devices. In some embodiments, memory controller 120 causes data to be written to and read from pages in a respective memory device or portion thereof according to one or more configuration parameters (e.g., storage density, ECC technique, and programming parameter(s)) indicated in a characterization vector for the respective memory device or portion thereof (e.g., a group of blocks, block, word line, or page). For example, while the respective memory device is configured in the first storage density, memory controller 120 or a component thereof (e.g., request handling module 218, Figure 2A) receives a command from the host (e.g., computer system 110, Figure 1) to write user data (sometimes also called write data) to a logical address or set of logical addresses. Continuing with this example, memory controller 120 uses logical-to-physical mapping 236 (Figure 2A) to map the logical address or set of logical addresses identified by the write command to physical addresses of one or more pages in the respective memory device or portion thereof. Continuing with this example, memory controller 120 selects a characterization vector associated with the respective memory device or portion thereof from characterization vector table 234 (Figures 2A-2B) based on the physical addresses so as to determine an ECC technique with which to encode the user data, programming parameter(s) for writing the user data to the one or more pages, and the first storage density associated with memory cells in the respective memory device or portion thereof. Continuing with this example, encoder 126 (Figure 1) encodes the user data according to the determined ECC technique, and memory controller 120 or a component thereof (e.g., data read module 212, Figure 2A or data write module 214, Figure 2A) causes read/write circuitry in storage medium I/O 128 (Figure 1) or read/write circuitry 135 (Figure 1) to write the encoded user data (i.e., one or more codewords) to the one or more pages in the respective memory device or portion thereof according to the determined programming parameter(s).

[0057] In some embodiments, while the one or more memory devices are configured in the first storage density, memory cells comprising the one or more memory devices each have (504) a storage capacity of N bits of data, where N is greater than zero. For example, when the one or more memory devices comprising storage device 130 are configured in the first storage density, the memory cells of the one or more memory devices are configured as X3 memory cells, and thus have a storage capacity of 3 bits of data per memory cell.

[0058] In some embodiments, after determining the initial storage capacities of the one or more memory devices, the memory controller reports (506), to a host system, the initial storage capacity of at least one of the one or more memory devices. In some embodiments, after determining the initial storage capacities for each of the one or more memory devices associated with storage device 130, memory controller 120 or a component thereof (e.g., reporting module 219, Figure 2A) reports to the host (e.g., computer system 110, Figure 1) the storage capacity of at least one of the one or more memory devices associated with storage device 130. In some embodiments, reporting module 219 automatically reports the storage capacity of at least one of the one or more memory devices. In some embodiments, reporting module 219 reports the storage capacity of at least one of the one or more memory devices in response to a query from the host.

[0059] As a practical matter, it is noted that in some embodiments storage device 130 will have a preset or predetermined initial storage capacity, based on a presumption that most or all of the memory cells of the storage device will be configured with a preset storage density (e.g., 98% of the memory cells configured as X3 memory cells and 2% of the memory cells configured as X1 memory cells), and also a presumption that storage device 130 will have a preset amount of over provisioning (e.g., 10% over provisioning). In such embodiments, the initially reported storage capacity mentioned above is the preset initial storage capacity unless an overriding condition is detected, requiring that the initial storage capacity reported be lower than the preset storage capacity. Examples of overriding conditions are a larger number of unusable die, blocks or pages than expected, or a larger number of blocks or pages that need to be configured at a lower storage density (e.g., X1 instead of X3) than expected.

[0060] In some embodiments, the memory controller maintains (508) one or more status metrics for each of the one or more memory devices, where a respective status metric corresponds to a respective memory device's ability to retain data. In some embodiments, memory controller 120 or a component thereof (e.g., metric maintaining module 222, Figure 2A) maintains one or more status metrics for each of the one or more memory devices associated with storage device 130. In some embodiments, the one or more status metrics indicate a respective memory device's ability to retain data. For example, as the respective memory device ages, the one or more status metrics reflect the respective memory device's diminished ability to retain data (e.g., data read from the respective memory device includes more errors as the respective memory device ages). In some embodiments, the one or more

status metrics associated with a respective memory device are stored in a characterization vector corresponding to the respective memory device. In some embodiments, the one or more status metrics stored in the characterization vector for the respective memory device include: (a) a bytes written field 260 indicating a number of bytes of data written to pages in the respective memory device; (b) a PE cycle field 262 indicating a current count of the number of PE cycles performed on the respective memory device; (c) an BER field 264 indicating a number of errors included in a codeword read from pages of the respective memory device; and (d) other usage information 266 indicating the health, performance, and/or endurance of the respective memory device.

[0061] In some embodiments, memory controller 120 or a component thereof (e.g., metric maintaining module 222, Figure 2A) maintains a combined status metric for each of the one or more memory devices associated with storage device 130. For example, the combined status metric indicates an amount of wear remaining for a respective memory device. In some embodiments, the combined status metric associated with a respective memory device is stored in a characterization vector corresponding to the respective memory device. In some embodiments, combined status metric 258 is associated with the output of a predefined algorithm (e.g., computed by metric maintaining module 222, Figure 2A) that takes into account one or more usage parameters and/or device performance parameters associated with the respective memory device. For example, the predefined algorithm incorporates one or more of: (a) a number of bytes written to the respective memory device; (b) a count of PE cycles performed on respective memory device; (c) a BER for codewords read from page(s) of the respective memory device; and (d) other usage or performance information associated with the respective memory device.

[0062] In some embodiments, the memory controller maintains (510) one or more metrics for each memory portion of the plurality of memory portions of the respective memory device, where a respective status metric for a respective memory portion corresponds to the respective memory portion's ability to retain data. In some embodiments, memory controller 120 or a component thereof (e.g., metric maintaining module 222, Figure 2A) maintains one or more status metrics for each memory portion (of the plurality of memory portions) of the one or more memory devices associated with storage device 130. In some embodiments, the one or more status metrics indicate a respective memory portion's ability to retain data. For example, as the respective memory portion ages, the one or more status metrics reflect the respective memory portion's diminished ability to retain data (e.g.,

data read from the respective memory portion includes more errors as the respective memory portion ages). In some embodiments, the one or more status metrics associated with a respective memory portion are stored in a characterization vector corresponding to the respective memory portion. In some embodiments, the one or more status metrics stored in the characterization vector for the respective memory portion include: (a) a bytes written field 260 indicating a number of bytes of data written to pages in the respective memory portion; (b) a PE cycle field 262 indicating a current count of the number of PE cycles performed on the respective memory portion; (c) an BER field 264 indicating a number of errors included in a codeword read from pages of the respective memory portion; and (d) other usage or performance information 266 indicating the health, performance, and/or endurance of the respective memory portion.

[0063] In some embodiments, memory controller 120 or a component thereof (e.g., metric maintaining module 222, Figure 2A) maintains a combined status metric for each respective memory portion of the one or more memory devices associated with storage device 130. For example, the combined status metric indicates an amount of wear remaining for a respective memory portion. In some embodiments, the combined status metric associated with a respective memory portion is stored in a characterization vector corresponding to the respective memory portion. In some embodiments, combined status metric 258 is associated with the output of a predefined algorithm (e.g., computed by metric maintaining module 222, Figure 2A) that takes into account one or more usage and/or performance parameters associated with the respective memory portion. For example, the predefined algorithm incorporates one or more of: (a) a number of bytes written to the respective memory portion; (b) a count of PE cycles performed on respective memory portion; (c) a BER for codewords read from page(s) of the respective memory portion; and (d) other usage or performance information associated with the respective memory portion.

[0064] In some embodiments, the memory controller receives (512) a query, and, in response to receiving the query, the memory controller reports at least one of the one or more status metrics for at least one of the one or more memory devices. In some embodiments, the query is received from the host (e.g., computer system 100, Figure 1), or a process internal to data storage system 100, or an accelerator module associated with storage device 130. For example, the host system sends a query to memory controller 120 for the status of a logical address or set of logical addresses. Continuing with this example, memory controller 120 uses logical-to-physical mapping 236 (Figure 2A) to map the logical address or set of logical

addresses identified by the query to physical addresses of memory devices or memory portions in storage device 130. Continuing with this example, memory controller 120 selects a characterization vector associated with a respective memory device or portion thereof from characterization vector table 234 (Figures 2A-2B) based on the physical addresses so as to determine current status metric(s) for the respective memory device or portion thereof. Thereafter, in this example, memory controller 120 or a component thereof (e.g., reporting module 219, Figure 1) reports at least one of the current status metrics for the respective memory device or portion thereof to the host. For example, reporting module 219 reports the combined status metric (i.e., the wear remaining) for a respective memory device identified in the query to the host.

[0065] In some embodiments, the memory controller (514): receives a memory command to write write data to the storage system; in response to receiving the memory command, writes the write data to the one or more memory devices; and, after writing the write data, sends a notification acknowledging writing the write data, where the notification also includes at least one of the one or more status metrics for at least one of the one or more memory devices. For example, memory controller 120 or a component thereof (e.g., request handling module 218, Figure 2A) receives a command from the host (e.g., computer system 110, Figure 1) to write user data (sometimes also called write data) to a logical address or set of logical addresses. Continuing with this example, memory controller 120 uses logical-to-physical mapping 236 (Figure 2A) to map the logical address or set of logical addresses identified by the write command to physical addresses of memory devices or portions in storage device 130. Continuing with this example, memory controller 120 selects a characterization vector associated with a respective memory device or portion thereof from characterization vector table 234 (Figures 2A-2B) based on the physical addresses so as to determine an ECC technique with which to encode the write data, programming parameter(s) for writing the write data to the one or more pages, the storage density associated with memory cells in the respective memory device or portion thereof, and current status metric(s) for the respective memory device or portion thereof. Alternatively, memory controller 120 selects a characterization vector associated with a respective memory device or portion thereof so as to determine one or more of the aforementioned parameters and/or metrics.

[0066] Continuing with this example, encoder 126 (Figure 1) encodes the write data according to the determined ECC technique, and memory controller 120 or a component thereof (e.g., data read module 212, Figure 2A or data write module 214, Figure 2A) causes

read/write circuitry in storage medium I/O 128 (Figure 1) or read/write circuitry 135 (Figure 1) to write the encoded user data (i.e., a codeword) to the respective memory device or portion thereof according to the determined programming parameter(s). Thereafter, in this example, memory controller 120 or a component thereof (e.g., reporting module 219, Figure 1) acknowledges successful completion of the write command and also reports at least one of the current status metric(s) for the respective memory device or portion thereof to the host.

[0067] The memory controller detects (516) a trigger condition as to at least one memory portion of a respective device of the one or more memory devices. In some embodiments, memory controller 120 or a component thereof (e.g., trigger detection module 224, Figure 2A) detects a trigger condition as to a memory portion of a respective memory device or as to a respective memory device associated with storage device 130. For example, trigger detection module 224 automatically detects a trigger condition as to at least one memory portion of a respective memory device associated with storage device 130.

[0068] In some embodiments, the trigger condition is detected by receiving (518) a command (e.g., from the host system) to recharacterize the at least one memory portion of the respective memory device. In some embodiments, memory controller 120 or a component thereof (e.g., trigger detection module 224, Figure 2A) detects a trigger condition as to a memory portion of a respective memory device or as to the respective memory device associated with storage device 130 in response to receiving a recharacterization command from the host (e.g., computer system 110, Figure 1) to recharacterize the memory portion of the respective memory device or as to the respective memory device. For example, the host sends the recharacterization command as to a respective memory device when the respective memory device enters a read-only mode (e.g., indicating that the respective memory device has reached end-of-life conditions). In another example, the host monitors the status metric(s) corresponding to the memory devices and/or memory portions thereof, and the host sends the recharacterization command as to a respective memory device or a respective memory portion when a status metric for the respective memory device or the respective memory portion falls below a predetermined threshold (e.g., the count of PE cycles performed on the respective memory device or the respective memory portion exceeds a predetermined count, or the number of bytes written to the respective memory device or the respective memory portion exceeds a predetermined number of bytes).

[0069] In some embodiments, the trigger condition is detected, in accordance with a determination (520) that at least one of the one or more status metrics for the at least one

memory portion of a respective device satisfies one or more predefined criteria, by detecting the trigger condition as to the at least one memory portion of a respective device. In some embodiments, memory controller 120 or a component thereof (e.g., trigger detection module 224, Figure 2A) detects a trigger condition as to a memory portion of a respective memory device or as to a respective memory device when at least one of the one or more status metrics corresponding to the memory portion of the respective memory device or the respective memory device satisfies one or more predefined criteria. In some embodiments, the one or more status metrics are stored in a characterization vector corresponding to the respective memory portion or the respective memory device. For example, memory controller 120 detects the trigger condition for a respective memory portion when the one or more status metrics corresponding to the respective memory portion indicate that a total number or average number of bytes written to pages in the respective memory portion exceeds a predetermined number of bytes, that a count of PE cycles performed on the respective memory portion exceeds a predetermined count, or that the value of a combined status metric for the respective memory portion is below a predefined recharacterization threshold. As such, continuing with this example, memory controller 120 triggers recharacterization of the respective memory portion from the first storage density to a second storage density, where the respective memory portion is able to store less data when configured in the second storage density.

[0070] For example, with reference to graph 320 of Figure 3, trigger detection module 224 detects the trigger condition as to a respective memory device at time 304 when the amount of wear remaining for the respective memory device is equal to zero (or other predefined threshold). In another example, with reference to graph 420 of Figure 4, trigger detection module 224 detects the trigger condition as to a respective memory device at time 404 when the amount of wear remaining for the respective memory device is equal to zero (or other predefined threshold). In one example, a memory portion is rated by the manufacturer to write up X bytes before a bit error rate associated with the memory portion becomes intolerable. When X bytes have been written to the memory portion, memory controller 120 detects the trigger as to the memory portion and recharacterizes the memory portion so as to be configured in a lower storage density. In another example, when Y% of X bytes (where Y% is less than 100%) have been written to the memory portion, memory controller 120 detects the trigger as to the memory portion and recharacterizes the memory portion so as to be configured in a lower storage density. In a further example, memory controller 120 detects

the trigger as to a memory portion when a combined status metric for the memory portion falls below a predefined recharacterization threshold, where the combined status metric includes additional history, performance, and/or health factors (e.g., BER and operating conditions) in addition to the number of bytes written to the memory portion.

[0071] In some embodiments, in response to detecting the trigger condition and prior to the recharacterizing, the memory controller migrates (522) data stored in the at least one memory portion of the respective memory device to another memory portion in the storage system. In some embodiments, after detecting the trigger condition as to the respective memory portion and prior to recharacterizing the respective memory portion, memory controller 120 or a component thereof (e.g., migration module 226, Figure 2A) migrates data stored in the respective memory portion. In some embodiments, migration module 226 automatically migrates the data stored by the respective memory portion prior to recharacterizing the portion without receiving a command from the host. In some embodiments, migration module 226 migrates the data stored by the respective memory portion after receiving a recharacterization command from the host. Alternatively, in some embodiments, the host (e.g., computer system 110, Figure 1) migrates the data stored in the respective memory portion instead of memory controller 120.

[0072] In some embodiments, migration is not required prior to recharacterization when data storage system 100 is configured to redundantly store data, for example using a RAID configuration or maintaining at least N replicable copies of the data (e.g., $N \geq 2$). In some embodiments, when data storage system 100 is not distributed or redundantly configured, data is migrated prior to recharacterization. In one example, the data stored by the respective memory portion is migrated to a different memory device associated with data storage system 100 than the memory device including the respective memory portion. In another example, the data stored by the respective memory portion is migrated to a different memory portion in the same memory device that includes the respective memory portion.

[0073] In response to detecting the trigger condition, the memory controller recharacterizes (524) the at least one memory portion of the respective memory device so as to be configured in a second storage density, where the at least one recharacterized memory portion of the respective memory device has a reduced storage capacity. In some embodiments, in response to detecting the trigger condition as to the at least one memory portion, memory controller 120 or a component thereof (e.g., recharacterization module 228, Figure 2A) recharacterizes the at least one memory portion of the respective memory device

so as to be configured in the second storage density. In some embodiments, after the recharacterizing, memory controller 120 or a component thereof (e.g., metric maintaining module 222, Figure 2A) updates the storage density field 252 of the characterization vector for the at least one memory portion to indicate that the at least one memory portion is configured in the second storage density.

[0074] In some embodiments, after the respective memory portion is recharacterized so as to be configured in the second storage density, memory cells in the respective memory portion each have (526) a storage capacity of M bits of data, where M is greater than zero and M is less than N. For example, when configured in the first storage density, memory cells of the at least one memory portion are configured to store N bits of data (e.g., X3). Continuing with this example, after recharacterizing the at least one memory portion so as to be configured in the second storage density, memory cells of the at least one memory portion are configured to store M bits of data (e.g., X2), where $N > M$.

[0075] In some embodiments, maintaining the endurance metric for each of the one or more memory devices includes (528), determining, in conjunction with the recharacterizing of the at least one memory portion of the respective memory device, an increased value for at least one of the one or more status metrics for the respective memory device. In some embodiments, after recharacterizing the at least one memory portion so as to be configured in the second storage density, at least one of the status metrics associated with the memory device corresponding to the at least one memory portion increases. For example, with reference to Figure 4, as blocks of the respective memory device are serially recharacterized in window 408, the wear remaining in graph 420 for the respective memory device increases. In some embodiments, after the recharacterizing, memory controller 120 or a component thereof (e.g., metric maintaining module 222, Figure 2A) updates one or more status metrics in the characterization vector for the respective memory device that includes the at least one memory portion to indicate that the at least one memory portion is configured in the second storage density. For example, metric maintaining module 222 increases the combined status metric 258 for the respective memory device to indicate a greater amount of wear remaining while the at least one memory portion of the respective memory device is configured in the second storage density.

[0076] In some embodiments, maintaining the endurance metric for each memory portion includes (530), determining, in conjunction with the recharacterizing of the at least one memory portion of the respective memory device, an increased value for at least one of

the one or more status metrics for the at least one memory portion of the respective memory device. In some embodiments, after the recharacterizing, memory controller 120 or a component thereof (e.g., metric maintaining module 222, Figure 2A) updates one or more status metrics in the characterization vector for the at least one memory portion to indicate that the at least one memory portion is configured in the second storage density.

Alternatively, in some embodiments, the combined status metric is insensitive to the current storage density. Instead, for example, when a write is performed to a memory portion while configured as X3, PE cycles field 262 in the characterization vector for the memory portion is incremented by 3, whereas when the write is performed to the memory portion while configured as X1, PE cycles field 262 in the characterization vector for the memory portion is incremented by 1.

[0077] After the recharacterizing, the memory controller determines (532) a revised storage capacity for the respective memory device. In some embodiments, after recharacterizing the at least one memory portion, memory controller 120 determines a revised storage capacity for the respective memory device that includes the at least one memory portion. In some embodiments, after determining the revised storage capacity for the respective memory device, memory controller 120 or a component thereof (e.g., metric maintaining module 222, Figure 2A) updates the storage capacity field 256 of the characterization vector for the respective memory device to indicate the revised storage capacity for the respective memory device. In Figure 3, for example, after the respective memory device is recharacterized at time 304, graph 330 shows that the advertised storage capacity for the respective memory device is reduced from amount 332 to amount 334. In one example, when the respective memory device is recharacterized from X3 to X2, the respective memory device's storage capacity decreases by approximately 33.33%. In another example, when the respective memory device is recharacterized from X2 to X1, the respective memory device's storage capacity decreases by approximately 50%.

[0078] In some embodiments, after determining the revised storage capacity for the respective memory device, the memory controller reports (534), to the host system, the revised storage capacity of the respective memory device. In some embodiments, after determining the revised storage capacity for the respective memory device including the at least one memory portion, memory controller 120 or a component thereof (e.g., reporting module 219, Figure 2A) reports to the host (e.g., computer system 110, Figure 1) the revised storage capacity of the respective memory device. In some embodiments, reporting module

219 automatically reports the revised storage capacity to the host. In some embodiments, reporting module 219 reports the revised storage capacity to the host in response to a query from the host, or alternatively, in response to a host command such as a read, write or erase command.

[0079] In some embodiments, while the at least one memory portion of the respective memory device is configured in the second storage density, memory controller 120 writes data to and reads data from the at least one memory portion of the respective memory device. For example, after recharacterizing the at least one memory portion, memory controller 120 causes data to be written to and read from pages in the at least one memory portion according to the configuration parameters (e.g., the second storage density, ECC technique, and programming parameter(s)) indicated in the updated characterization vector for the at least one memory portion.

[0080] Semiconductor memory devices include volatile memory devices, such as dynamic random access memory (“DRAM”) or static random access memory (“SRAM”) devices, non-volatile memory devices, such as resistive random access memory (“ReRAM”), electrically erasable programmable read only memory (“EEPROM”), flash memory (which can also be considered a subset of EEPROM), ferroelectric random access memory (“FRAM”), and magnetoresistive random access memory (“MRAM”), and other semiconductor elements capable of storing information. Each type of memory device may have different configurations. For example, flash memory devices may be configured in a NAND or a NOR configuration.

[0081] The memory devices can be formed from passive and/or active elements, in any combinations. By way of non-limiting example, passive semiconductor memory elements include ReRAM device elements, which in some embodiments include a resistivity switching storage element, such as an anti-fuse, phase change material, etc., and optionally a steering element, such as a diode, etc. Further by way of non-limiting example, active semiconductor memory elements include EEPROM and flash memory device elements, which in some embodiments include elements containing a charge storage region, such as a floating gate, conductive nanoparticles, or a charge storage dielectric material.

[0082] Multiple memory elements may be configured so that they are connected in series or so that each element is individually accessible. By way of non-limiting example, flash memory devices in a NAND configuration (NAND memory) typically contain memory

elements connected in series. A NAND memory array may be configured so that the array is composed of multiple strings of memory in which a string is composed of multiple memory elements sharing a single bit line and accessed as a group. Alternatively, memory elements may be configured so that each element is individually accessible (e.g., a NOR memory array). NAND and NOR memory configurations are exemplary, and memory elements may be otherwise configured.

[0083] The semiconductor memory elements located within and/or over a substrate may be arranged in two or three dimensions, such as a two dimensional memory structure or a three-dimensional memory structure.

[0084] In a two dimensional memory structure, the semiconductor memory elements are arranged in a single plane or a single memory device level. Typically, in a two dimensional memory structure, memory elements are arranged in a plane (e.g., in an x-z direction plane) which extends substantially parallel to a major surface of a substrate that supports the memory elements. The substrate may be a wafer over or in which the layer of the memory elements are formed or it may be a carrier substrate which is attached to the memory elements after they are formed. As a non-limiting example, the substrate may include a semiconductor such as silicon.

[0085] The memory elements may be arranged in the single memory device level in an ordered array, such as in a plurality of rows and/or columns. However, the memory elements may be arrayed in non-regular or non-orthogonal configurations. The memory elements may each have two or more electrodes or contact lines, such as bit lines and word lines.

[0086] A three-dimensional memory array is arranged so that memory elements occupy multiple planes or multiple memory device levels, thereby forming a structure in three dimensions (i.e., in the x, y and z directions, where the y direction is substantially perpendicular and the x and z directions are substantially parallel to the major surface of the substrate).

[0087] As a non-limiting example, a three-dimensional memory structure may be vertically arranged as a stack of multiple two dimensional memory device levels. As another non-limiting example, a three-dimensional memory array may be arranged as multiple vertical columns (e.g., columns extending substantially perpendicular to the major surface of the substrate, i.e., in the y direction) with each column having multiple memory elements in

each column. The columns may be arranged in a two dimensional configuration (e.g., in an x-z plane), resulting in a three-dimensional arrangement of memory elements with elements on multiple vertically stacked memory planes. Other configurations of memory elements in three dimensions can also constitute a three-dimensional memory array.

[0088] By way of non-limiting example, in a three-dimensional NAND memory array, the memory elements may be coupled together to form a NAND string within a single horizontal (e.g., x-z) memory device level. Alternatively, the memory elements may be coupled together to form a vertical NAND string that traverses across multiple horizontal memory device levels. Other three-dimensional configurations can be envisioned wherein some NAND strings contain memory elements in a single memory level while other strings contain memory elements which span through multiple memory levels. Three-dimensional memory arrays may also be designed in a NOR configuration and in a ReRAM configuration.

[0089] Typically, in a monolithic three-dimensional memory array, one or more memory device levels are formed above a single substrate. Optionally, the monolithic three-dimensional memory array may also have one or more memory layers at least partially within the single substrate. As a non-limiting example, the substrate may include a semiconductor such as silicon. In a monolithic three-dimensional array, the layers constituting each memory device level of the array are typically formed on the layers of the underlying memory device levels of the array. However, layers of adjacent memory device levels of a monolithic three-dimensional memory array may be shared or have intervening layers between memory device levels.

[0090] Then again, two dimensional arrays may be formed separately and then packaged together to form a non-monolithic memory device having multiple layers of memory. For example, non-monolithic stacked memories can be constructed by forming memory levels on separate substrates and then stacking the memory levels atop each other. The substrates may be thinned or removed from the memory device levels before stacking, but as the memory device levels are initially formed over separate substrates, the resulting memory arrays are not monolithic three-dimensional memory arrays. Further, multiple two dimensional memory arrays or three-dimensional memory arrays (monolithic or non-monolithic) may be formed on separate chips and then packaged together to form a stacked-chip memory device.

[0091] Associated circuitry is typically required for operation of the memory elements and for communication with the memory elements. As non-limiting examples, memory devices may have circuitry used for controlling and driving memory elements to accomplish functions such as programming and reading. This associated circuitry may be on the same substrate as the memory elements and/or on a separate substrate. For example, a controller for memory read-write operations may be located on a separate controller chip and/or on the same substrate as the memory elements.

[0092] The term “three-dimensional memory device” (or 3D memory device) is herein defined to mean a memory device having multiple memory layers or multiple levels (e.g., sometimes called multiple memory device levels) of memory elements, including any of the following: a memory device having a monolithic or non-monolithic 3D memory array, some non-limiting examples of which are described above; or two or more 2D and/or 3D memory devices, packaged together to form a stacked-chip memory device, some non-limiting examples of which are described above.

[0093] One of skill in the art will recognize that this invention is not limited to the two dimensional and three-dimensional exemplary structures described but cover all relevant memory structures within the spirit and scope of the invention as described herein and as understood by one of skill in the art.

[0094] It will be understood that, although the terms “first,” “second,” etc. may be used herein to describe various elements, these elements should not be limited by these terms. These terms are only used to distinguish one element from another. For example, a first trigger condition could be termed a second trigger condition, and, similarly, a second trigger condition could be termed a first trigger condition, without changing the meaning of the description, so long as all occurrences of the “first trigger condition” are renamed consistently and all occurrences of the “second trigger condition” are renamed consistently. The first trigger condition and the second trigger condition are both trigger conditions, but they are not the same trigger condition.

[0095] The terminology used herein is for the purpose of describing particular embodiments only and is not intended to be limiting of the claims. As used in the description of the embodiments and the appended claims, the singular forms “a,” “an” and “the” are intended to include the plural forms as well, unless the context clearly indicates otherwise. It will also be understood that the term “and/or” as used herein refers to and encompasses any

and all possible combinations of one or more of the associated listed items. It will be further understood that the terms “comprises” and/or “comprising,” when used in this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof.

[0096] As used herein, the term “if” may be construed to mean “when” or “upon” or “in response to determining” or “in accordance with a determination” or “in response to detecting,” that a stated condition precedent is true, depending on the context. Similarly, the phrase “if it is determined [that a stated condition precedent is true]” or “if [a stated condition precedent is true]” or “when [a stated condition precedent is true]” may be construed to mean “upon determining” or “in response to determining” or “in accordance with a determination” or “upon detecting” or “in response to detecting” that the stated condition precedent is true, depending on the context.

[0097] The foregoing description, for purpose of explanation, has been described with reference to specific implementations. However, the illustrative discussions above are not intended to be exhaustive or to limit the claims to the precise forms disclosed. Many modifications and variations are possible in view of the above teachings. The implementations were chosen and described in order to best explain principles of operation and practical applications, to thereby enable others skilled in the art.

What is claimed is:

1. A method of operation in a storage system that comprises a memory controller and one or more memory devices, each with a plurality of memory portions, the method comprising:

at the memory controller:

determining an initial storage capacity for each of the one or more memory devices, wherein the one or more memory devices are configured in a first storage density;

detecting a trigger condition as to at least one memory portion of a respective memory device of the one or more memory devices;

in response to detecting the trigger condition, recharacterizing the at least one memory portion of the respective memory device so as to be configured in a second storage density, wherein the at least one recharacterized memory portion of the respective memory device has a reduced storage capacity; and

after the recharacterizing, determining a revised storage capacity for the respective memory device.

2. The method of claim 1, further comprising:

in response to detecting the trigger condition and prior to the recharacterizing, migrating data stored in the at least one memory portion of the respective memory device to another memory portion in the storage system.

3. The method of any of claims 1-2, wherein detecting the trigger condition comprises receiving a command to recharacterize the at least one memory portion of the respective memory device.

4. The method of any of claims 1-3, further comprising:

maintaining one or more status metrics for each of the one or more memory devices, wherein a respective status metric corresponds to a respective memory device's ability to retain data.

5. The method of claim 4, wherein maintaining the one or more status metrics for each of the one or more memory devices includes, determining, in conjunction with the recharacterizing of the at least one memory portion of the respective memory device, an

increased value for at least one of the one or more status metrics for the respective memory device.

6. The method of any of claims 4-5, further comprising:
receiving a query; and
in response to receiving the query, reporting at least one of the one or more status metrics for at least one of the one or more memory devices.
7. The method of any of claims 4-6, further comprising:
receiving a memory command to write write data to the storage system;
in response to receiving the memory command, writing the write data to the one or more memory devices; and
after writing the write data, sending a notification acknowledging writing the write data, wherein the notification also includes at least one of the one or more status metrics for at least one of the one or more memory devices.
8. The method of any of claims 1-3, further comprising:
maintaining one or more metrics for each memory portion of the plurality of memory portions of the respective memory device, wherein a respective status metric for a respective memory portion corresponds to the respective memory portion's ability to retain data.
9. The method of claim 8, wherein maintaining the one or more status metrics for each memory portion includes, determining, in conjunction with the recharacterizing of the at least one memory portion of the respective memory device, an increased value for at least one of the one or more status metrics for the at least one memory portion of the respective memory device.
10. The method of any of claims 8-9, wherein detecting the trigger condition comprises:
in accordance with a determination that at least one of the one or more status metrics for the at least one memory portion of a respective device satisfies one or more predefined criteria, detecting the trigger condition as to the at least one memory portion of a respective device.

11. The method of any of claims 1-10, further comprising:
 - after determining the initial storage capacities of the one or more memory devices, reporting, to a host system, the initial storage capacity of at least one of the one or more memory devices; and
 - after determining the revised storage capacity for the respective memory device, reporting, to the host system, the revised storage capacity of the respective memory device.
12. The method of any of claims 1-11, wherein, while the one or more memory devices are configured in the first storage density, memory cells comprising the one or more memory devices have a storage capacity of N bits of data, wherein N is greater than zero, and
 - wherein, after the respective memory portion is recharacterized so as to be configured in the second storage density, the memory cells comprising the respective memory portion each have a storage capacity of M bits of data, wherein M is greater than zero and M is less than N.
13. The method of any of claims 1-12, wherein the one or more memory devices are flash memory devices, and
 - wherein each portion of the plurality of memory portions of the respective memory device comprises one of the set of: one die, one erase block, and one page.
14. The method of any of claims 1-13, wherein the one or more memory devices comprises one or more three-dimensional (3D) memory devices, each with a 3D array of memory cells, and circuitry associated with operation of memory elements in the one or more 3D memory devices.
15. The method of claim 14, wherein the circuitry and one or more memory elements in a respective 3D memory device, of the one or more 3D memory devices, are on the same substrate.
16. A storage system, comprising:
 - a storage device including one or more memory devices, each with a plurality of memory portions; and
 - a memory controller with one or more processors and memory storing one or more programs to be executed by the one or more processors, the one or more programs comprising instructions for:

determining an initial storage capacity for each of the one or more memory devices, wherein the one or more memory devices are configured in a first storage density;

detecting a trigger condition as to at least one memory portion of a respective memory device of the one or more memory devices;

in response to detecting the trigger condition, recharacterizing the at least one memory portion of the respective memory device so as to be configured in a second storage density, wherein the at least one recharacterized memory portion of the respective memory device has a reduced storage capacity; and

after the recharacterizing, determining a revised storage capacity for the respective memory device.

17. The storage system of claim 16, wherein detecting the trigger condition comprises receiving a command to recharacterize the at least one memory portion of the respective memory device.

18. The storage system of claim 16, wherein the one or more programs further comprise instructions for:

maintaining one or more metrics for each memory portion of the plurality of memory portions of the respective memory device, wherein a respective status metric for a respective memory portion corresponds to the respective memory portion's ability to retain data.

19. The storage system of claim 18, wherein maintaining the one or more status metrics for each memory portion includes, determining, in conjunction with the recharacterizing of the at least one memory portion of the respective memory device, an increased value for at least one of the one or more status metrics for the at least one memory portion of the respective memory device.

20. The storage system of claim 18, wherein detecting the trigger condition comprises: in accordance with a determination that at least one of the one or more status metrics for the at least one memory portion of a respective device satisfies one or more predefined criteria, detecting the trigger condition as to the at least one memory portion of a respective device.

21. The storage system of claim 16, wherein the one or more programs comprise instructions for performing or controlling performance of any of the methods of claims 2-15.

22. A non-transitory computer readable storage medium storing one or more programs, the one or more programs comprising instructions, which, when executed by a memory controller with one or more processors, cause the memory controller to perform operations comprising:

determining an initial storage capacity for each of one or more memory devices, wherein the one or more memory devices are configured in a first storage density and the one or more memory devices each include a plurality of memory portions;

detecting a trigger condition as to at least one memory portion of a respective memory device of the one or more memory devices;

in response to detecting the trigger condition, recharacterizing the at least one memory portion of the respective memory device so as to be configured in a second storage density, wherein the at least one recharacterized memory portion of the respective memory device has a reduced storage capacity; and

after the recharacterizing, determining a revised storage capacity for the respective memory device.

23. The non-transitory computer readable storage medium of claim 22, wherein detecting the trigger condition comprises receiving a command to recharacterize the at least one memory portion of the respective memory device.

24. The non-transitory computer readable storage medium of claim 22, wherein the instructions cause the memory controller to perform operations further comprising:

maintaining one or more metrics for each memory portion of the plurality of memory portions of the respective memory device, wherein a respective status metric for a respective memory portion corresponds to the respective memory portion's ability to retain data.

25. The non-transitory computer readable storage medium of claim 24, wherein maintaining the one or more status metrics for each memory portion includes, determining, in conjunction with the recharacterizing of the at least one memory portion of the respective memory device, an increased value for at least one of the one or more status metrics for the at least one memory portion of the respective memory device.

26. The non-transitory computer readable storage medium of claim 24, wherein detecting the trigger condition comprises:

in accordance with a determination that at least one of the one or more status metrics for the at least one memory portion of a respective device satisfies one or more predefined criteria, detecting the trigger condition as to the at least one memory portion of a respective device.

27. A non-transitory computer readable storage medium storing one or more programs, the one or more programs comprising instructions, which, when executed by a memory controller with one or more processors, cause the memory controller to perform or control performance of any of the methods of claims 1-15.

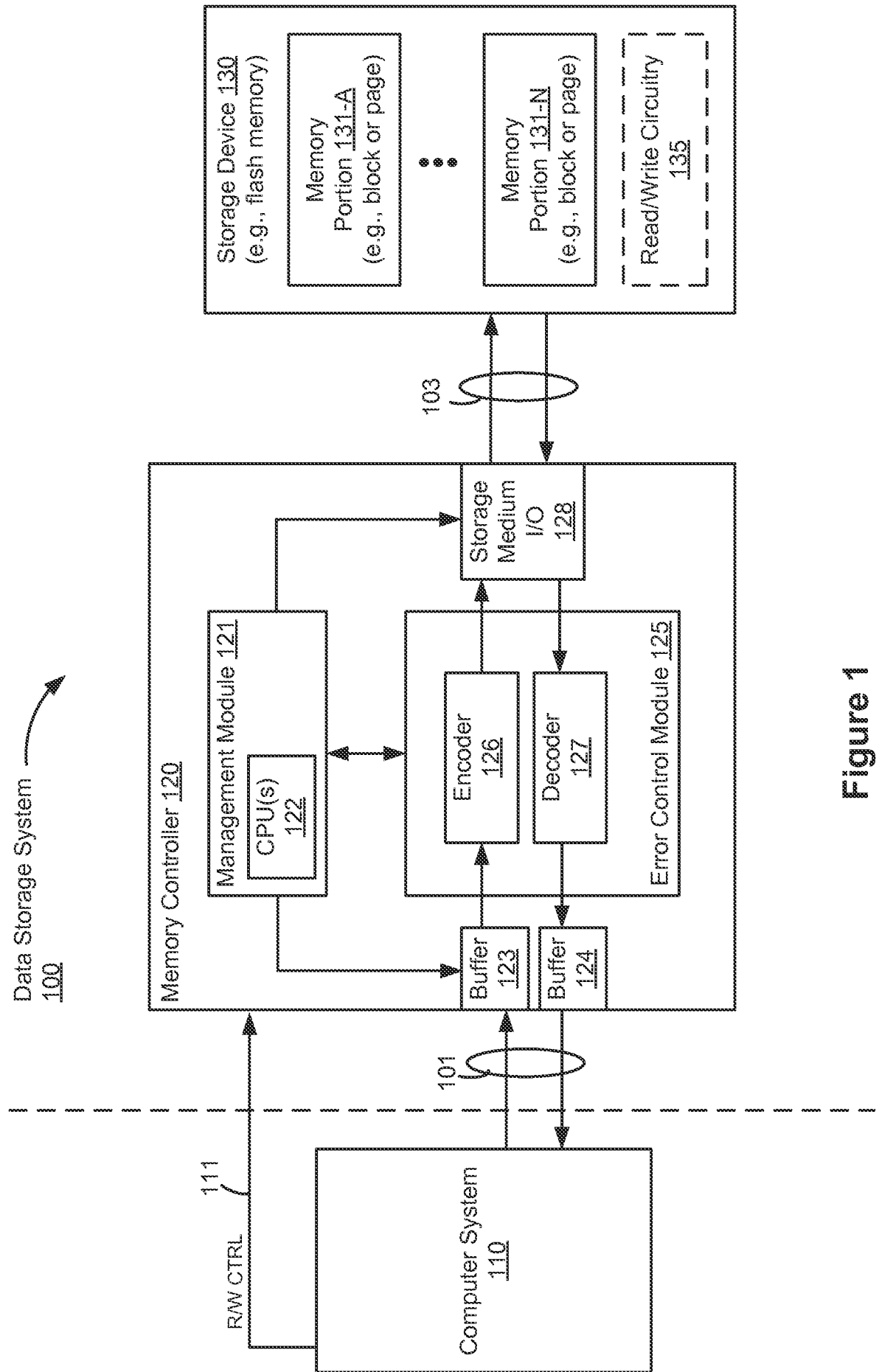


Figure 1

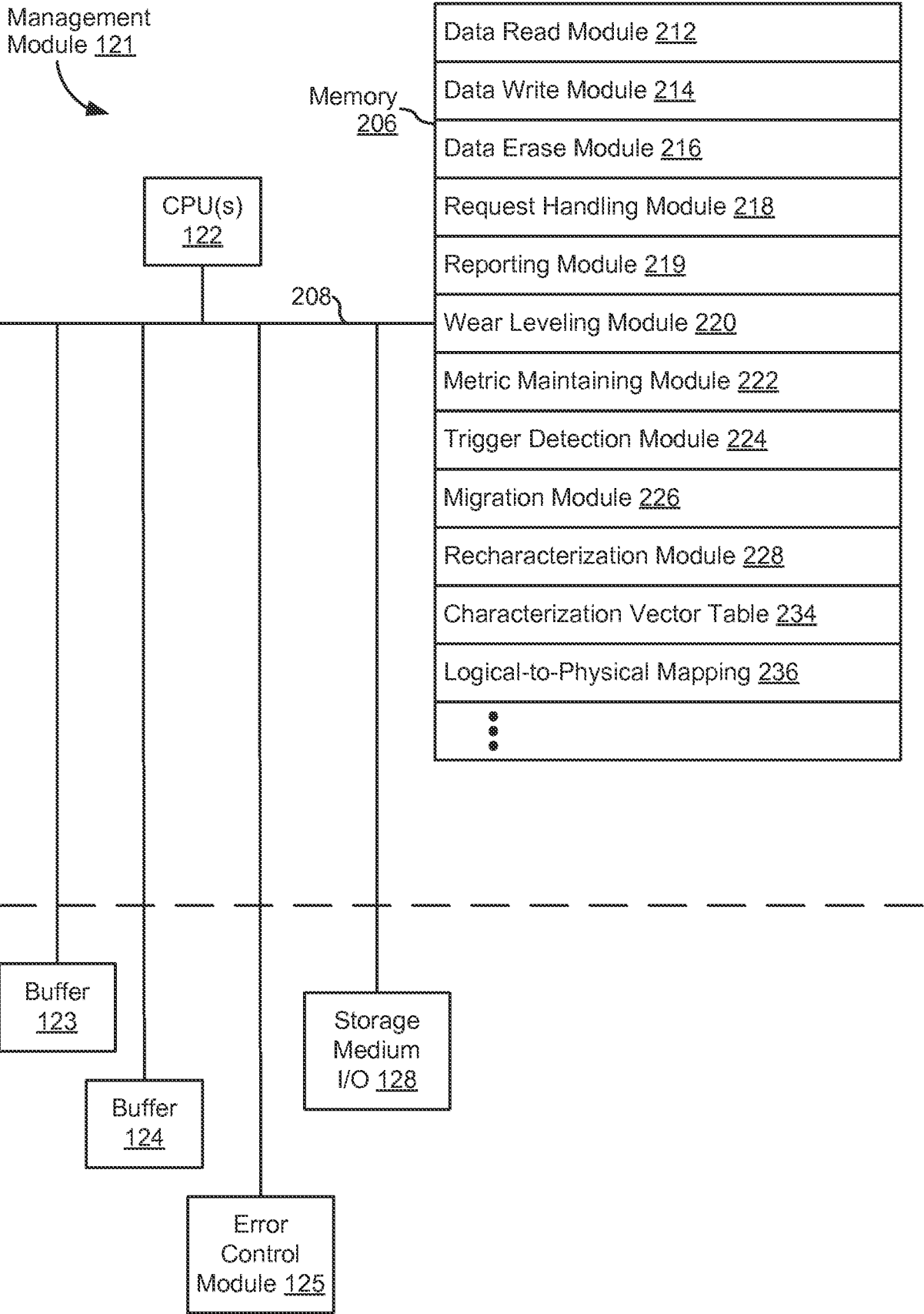


Figure 2A

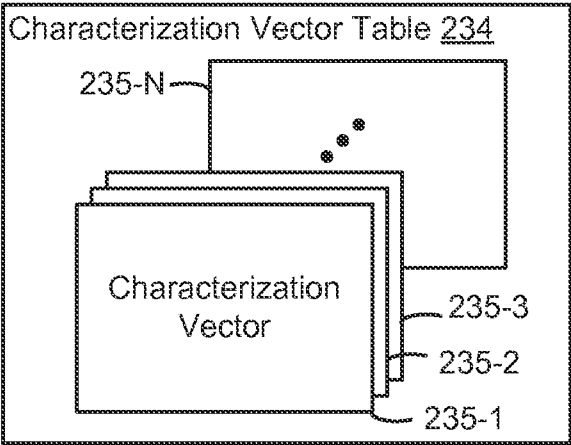


Figure 2B

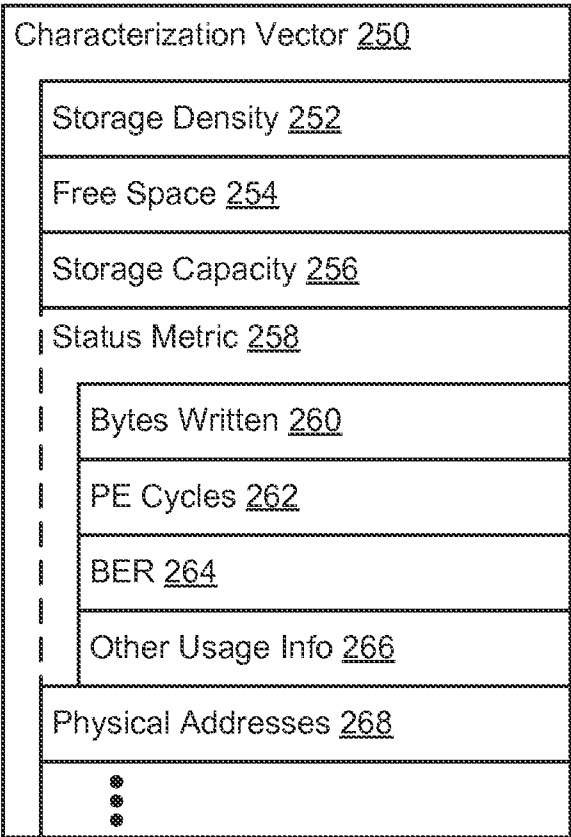


Figure 2C

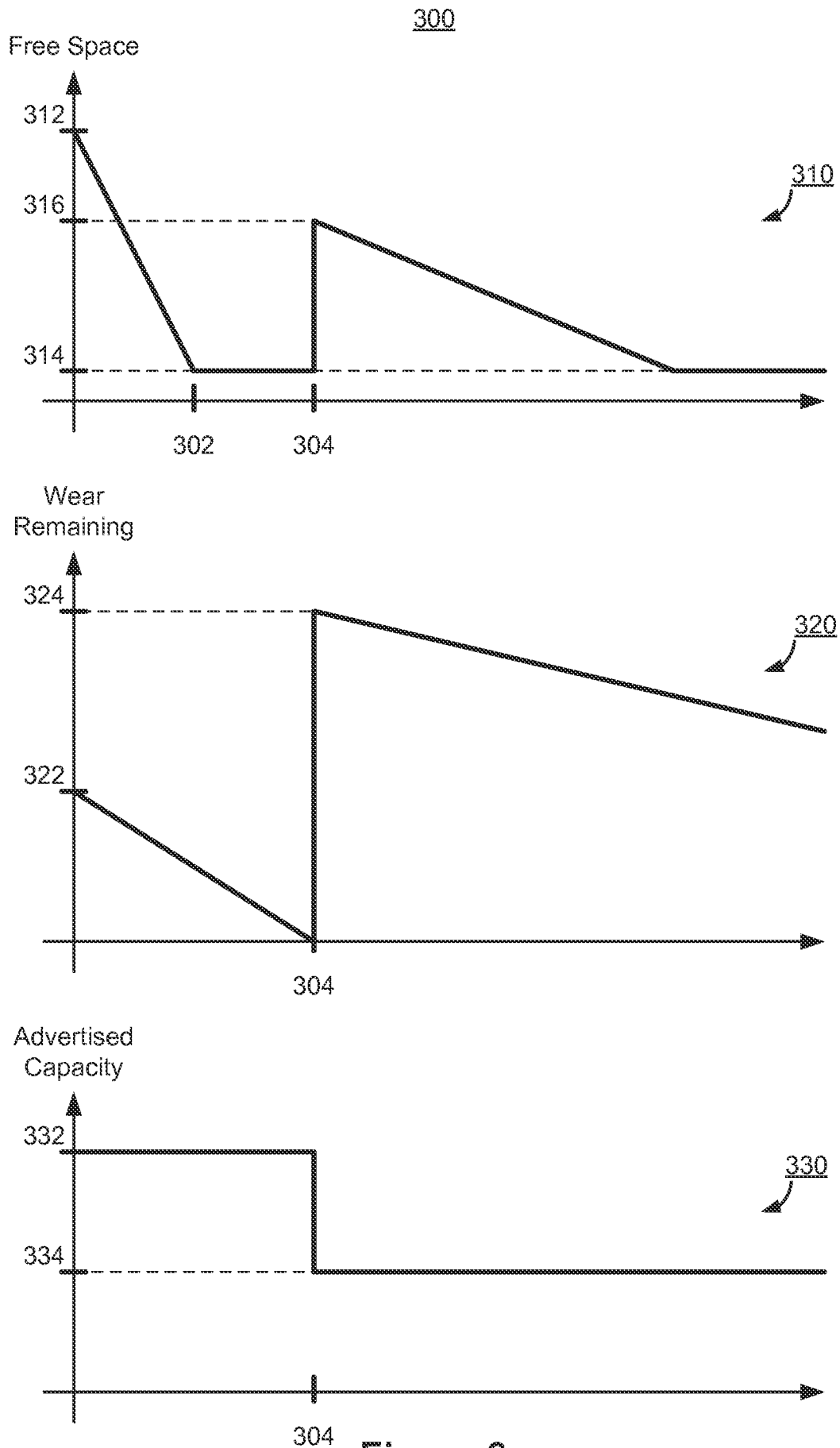
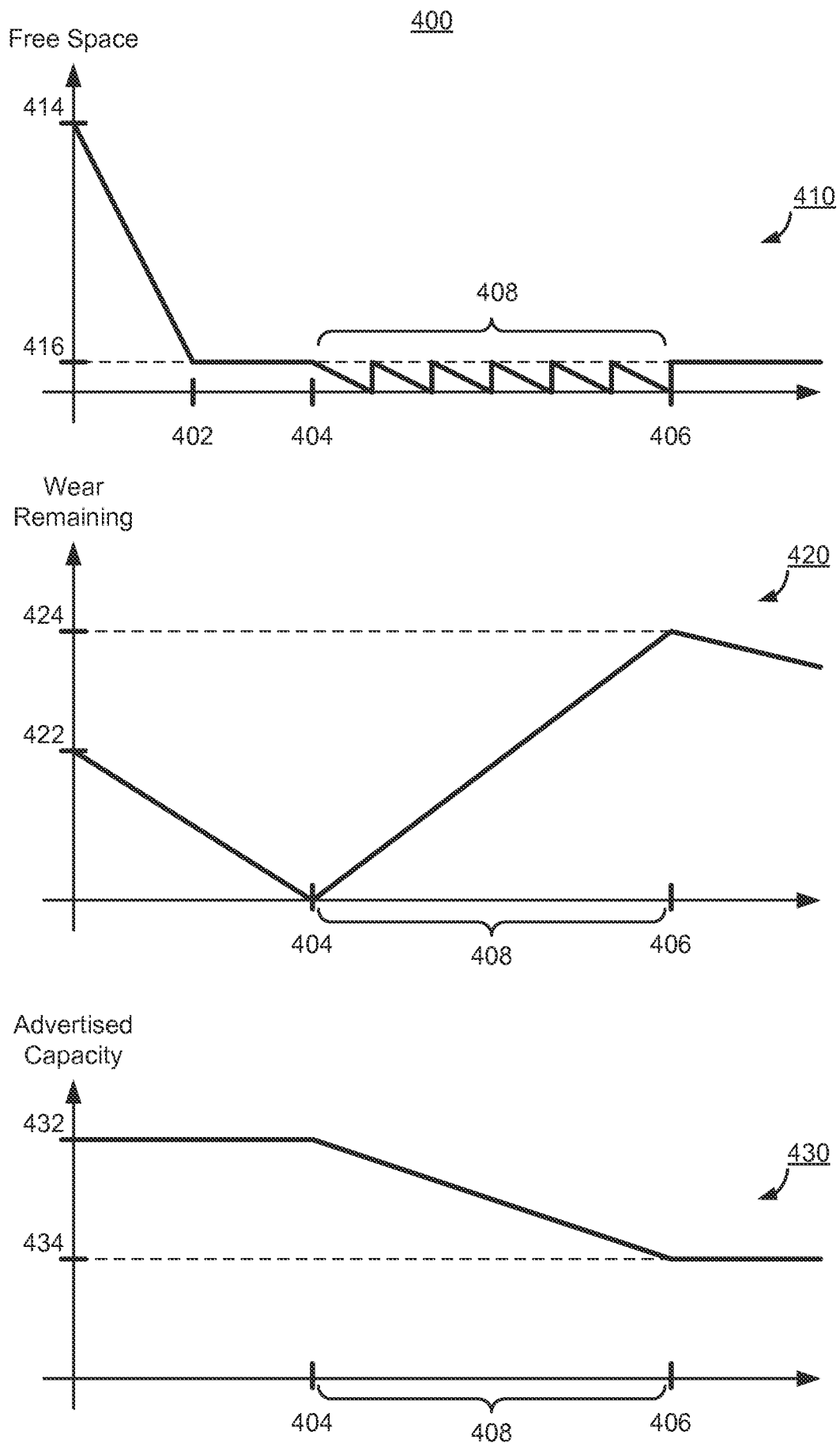


Figure 3

**Figure 4**

500

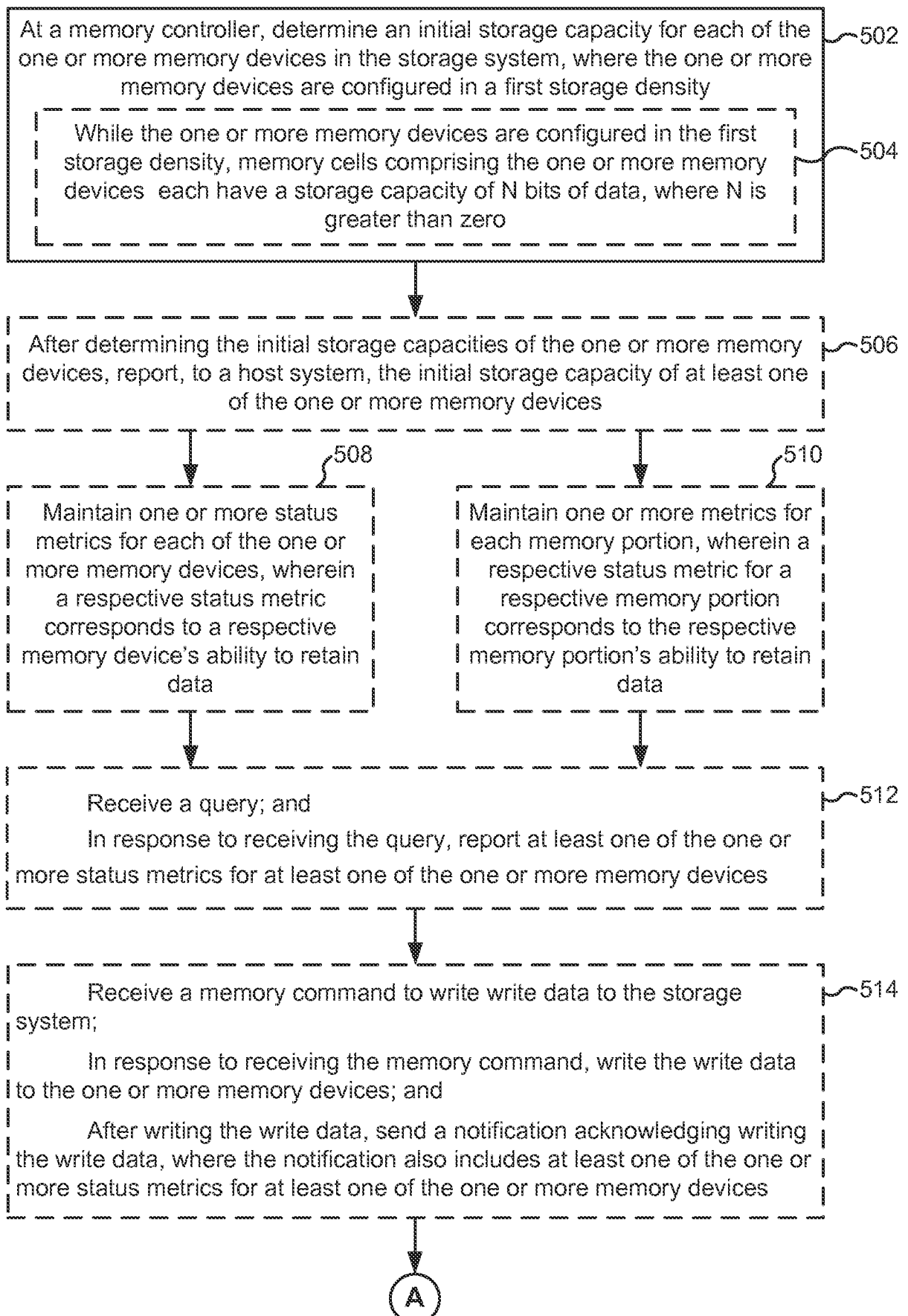
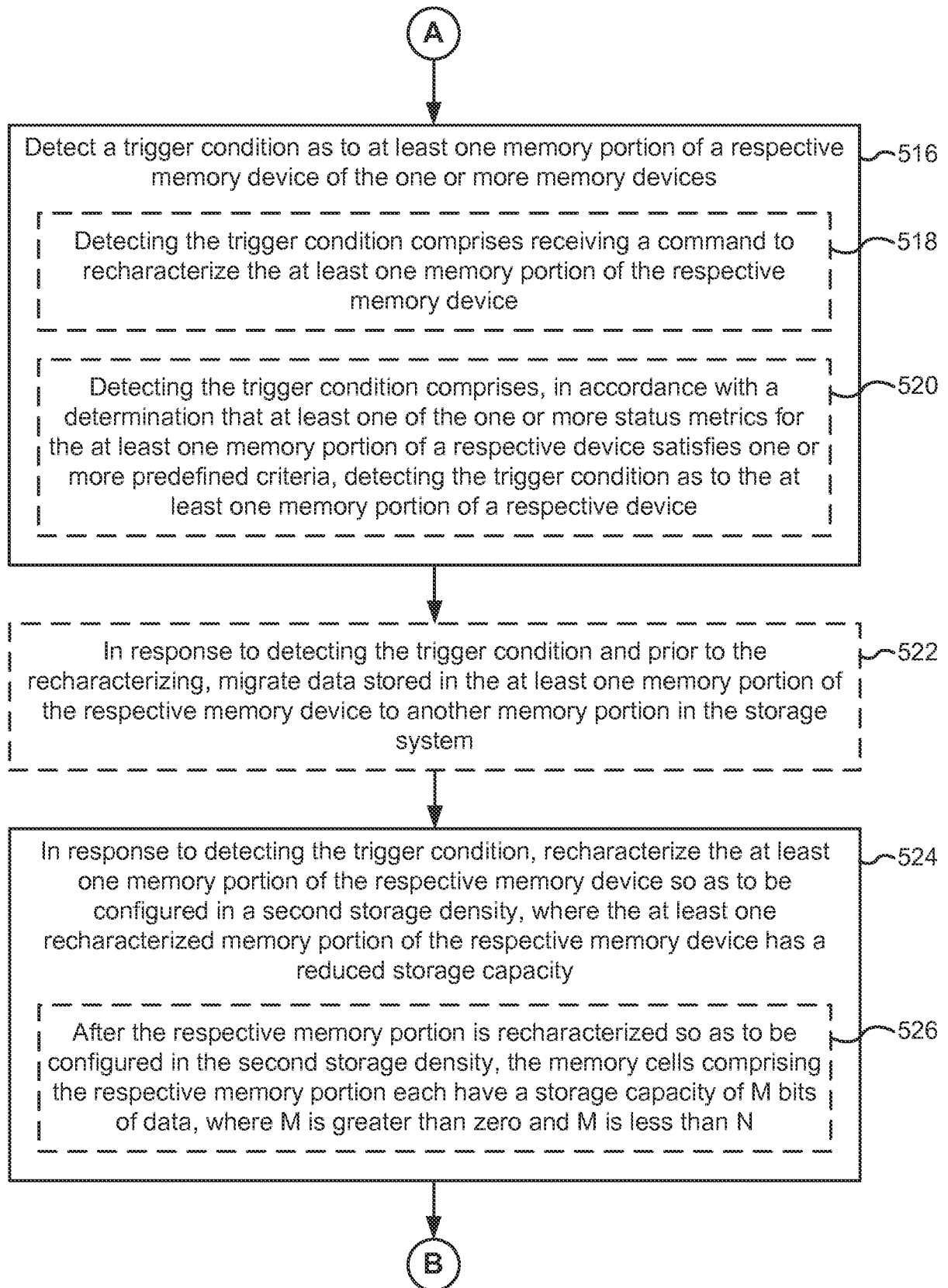
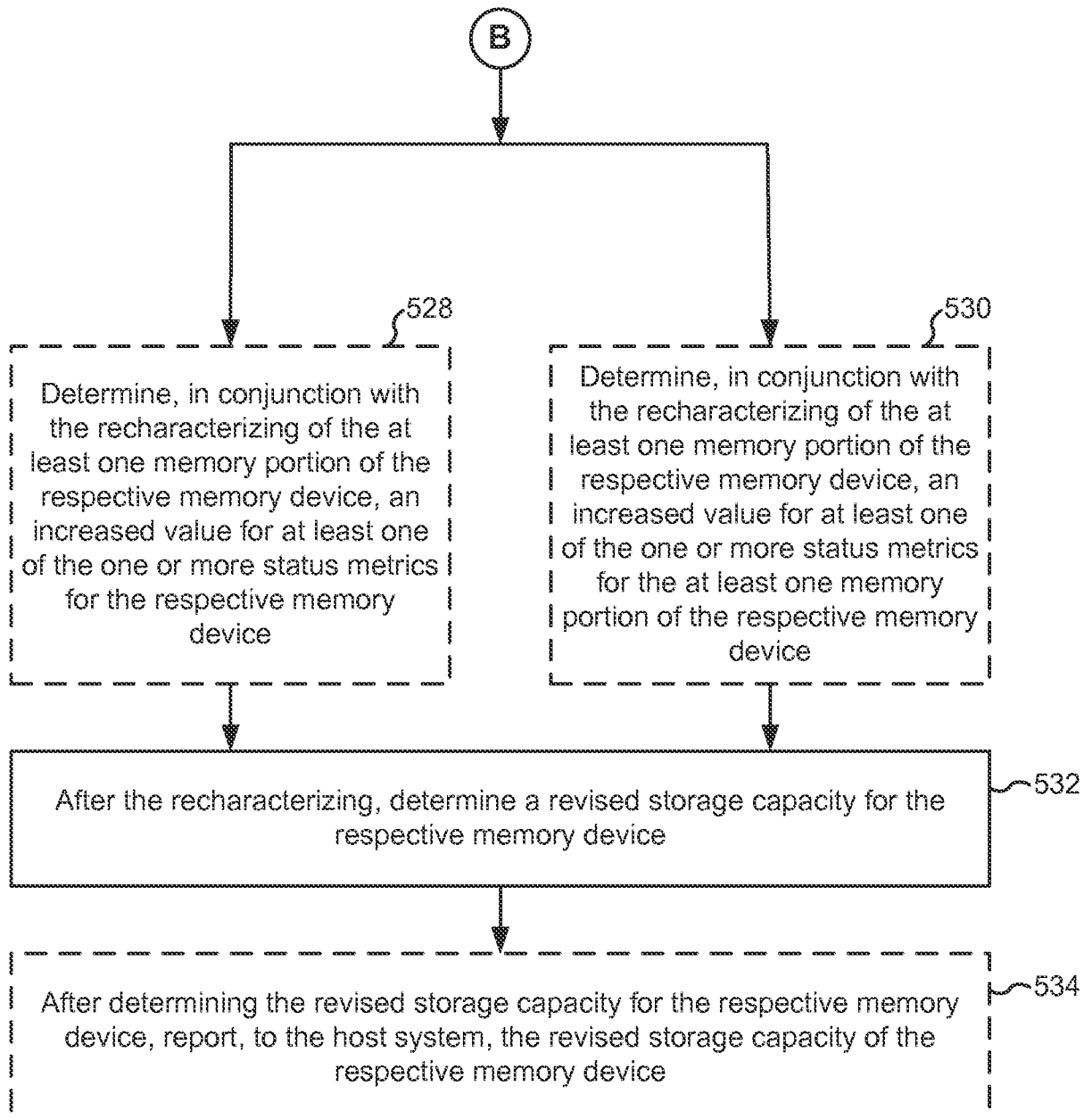


Figure 5A

**Figure 5B**

**Figure 5C**

INTERNATIONAL SEARCH REPORT

International application No
PCT/US2015/030847

A. CLASSIFICATION OF SUBJECT MATTER
INV. G11C13/00 G11C16/34 G11C11/56
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)
G11C

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)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2012/023285 A1 (KIM JIN-KI [CA]) 26 January 2012 (2012-01-26) paragraph [0052] - paragraph [0069]; figures 8a, 8b, 9-13 paragraph [0076] - paragraph [0084]; figure 15 paragraph [0045] - paragraph [0050]; figures 5-7 paragraph [0002] - paragraph [0009]; figure 1 -----	1-27
X	US 2010/008140 A1 (LEE CHANG-HYUN [KR]) 14 January 2010 (2010-01-14) paragraph [0048] - paragraph [0051]; figure 7 paragraph [0032] - paragraph [0035]; figure 2 ----- -/--	1-27



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

"E" earlier application or patent 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

10 July 2015

Date of mailing of the international search report

23/07/2015

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Balaguer López, J

INTERNATIONAL SEARCH REPORT

International application No

PCT/US2015/030847

C(Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2009/046509 A1 (ANNAVAJJHALA RAVI [US] ET AL) 19 February 2009 (2009-02-19) paragraph [0032] - paragraph [0036]; figures 4,5 -----	1-27
A	US 2008/112222 A1 (SHIRAKAWA MASANOBU [JP]) 15 May 2008 (2008-05-15) paragraph [0069] - paragraph [0092]; figures 6-9 -----	1-27
A	US 2008/235432 A1 (CHEN MING-DAR [TW] ET AL) 25 September 2008 (2008-09-25) paragraph [0030] - paragraph [0083]; figures 3-9 -----	1-27

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2015/030847

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2012023285	A1	26-01-2012	CA 2675565 A1 21-08-2008
			CN 101617372 A 30-12-2009
			EP 2126923 A1 02-12-2009
			EP 2490224 A2 22-08-2012
			ES 2477493 T3 17-07-2014
			JP 5421127 B2 19-02-2014
			JP 2010518546 A 27-05-2010
			JP 2013229100 A 07-11-2013
			KR 20090077840 A 15-07-2009
			KR 20090081426 A 28-07-2009
			KR 20130098439 A 04-09-2013
			KR 20140013079 A 04-02-2014
			KR 20140146229 A 24-12-2014
			TW 200845016 A 16-11-2008
			US 2008198651 A1 21-08-2008
			US 2010174854 A1 08-07-2010
			US 2012023285 A1 26-01-2012
			US 2013141973 A1 06-06-2013
			US 2014010019 A1 09-01-2014
			WO 2008098363 A1 21-08-2008
US 2010008140	A1	14-01-2010	CN 101625894 A 13-01-2010
			KR 20100007191 A 22-01-2010
			US 2010008140 A1 14-01-2010
US 2009046509	A1	19-02-2009	NONE
US 2008112222	A1	15-05-2008	JP 2008123330 A 29-05-2008
			KR 20080043711 A 19-05-2008
			US 2008112222 A1 15-05-2008
US 2008235432	A1	25-09-2008	CN 101271379 A 24-09-2008
			CN 101271380 A 24-09-2008
			CN 101271730 A 24-09-2008
			TW 200839510 A 01-10-2008
			TW 200839768 A 01-10-2008
			TW 200839773 A 01-10-2008
			US 2008235432 A1 25-09-2008
			US 2008235433 A1 25-09-2008
			US 2008235468 A1 25-09-2008
			US 2011246709 A1 06-10-2011