



- (51) **International Patent Classification:**
G06F 12/02 (2006.01) G06F 3/06 (2006.01)
G06F 13/16 (2006.01)
- (21) **International Application Number:**
PCT/US2017/021071
- (22) **International Filing Date:**
7 March 2017 (07.03.2017)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
15/084,187 29 March 2016 (29.03.2016) US
- (71) **Applicant:** MICRON TECHNOLOGY, INC. [US/US];
8000 South Federal Way Mailstop 525, Boise, Idaho
83707 (US).
- (72) **Inventor:** SZUBBOCSEV, Zoltan; Schinnerer Str 14B,
85778 Heimhausen (DE).
- (74) **Agents:** GUTKE, Steven W. et al.; 230 South 500 East,
Suite 300, Salt Lake City, Utah 84102 (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, DJ, 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, KH, KN,
KP, KR, KW, 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).

Declarations under Rule 4.17:

- as to applicant's entitlement to apply for and be granted a
patent (Rule 4.17(ii))
- as to the applicant's entitlement to claim the priority of the
earlier application (Rule 4.17(iii))

[Continued on next page]

(54) **Title:** MEMORY DEVICES INCLUDING DYNAMIC SUPERBLOCKS, AND RELATED METHODS AND ELECTRONIC SYSTEMS

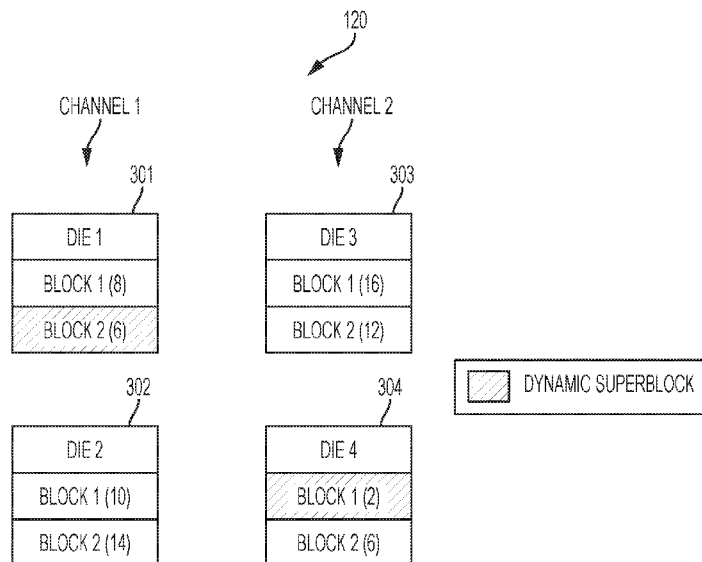


FIG. 3B

(57) **Abstract:** A memory device includes a memory array having non-volatile memory cells, and a memory controller configured to dynamically construct a superblock during each garbage collection process based, at least in part, on an amount of valid data present in each physical block of the memory array. Another memory device includes physical blocks of memory cells and a memory controller configured to construct a new superblock dynamically each time garbage collection occurs for the physical blocks regardless of whether any physical blocks are determined to be bad. Additional methods for managing operation of a memory device and related electronic systems are also described.

WO 2017/172293 A1

Published:

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

- 1 -

MEMORY DEVICES INCLUDING DYNAMIC SUPERBLOCKS, AND RELATED METHODS AND ELECTRONIC SYSTEMS

PRIORITY CLAIM

5 This application claims the benefit of the filing date of United States Patent Application Serial No. 15/084,187, filed March 29, 2016, for “MEMORY DEVICES INCLUDING DYNAMIC SUPERBLOCKS, AND RELATED METHODS AND ELECTRONIC SYSTEMS.”

TECHNICAL FIELD

10

The disclosure, in various embodiments, relates generally to the field of memory device management. More specifically, the disclosure relates to memory devices including flash translation layers constructing dynamic superblocs for garbage collection, and to related methods and electronic systems.

15

BACKGROUND

Non-volatile memory is commonly used for mass storage of data (e.g., music, videos, photos, and other data), such as within consumer electronic devices. Flash memory (e.g., NAND Flash) has become ubiquitous as non-volatile storage used within mobile consumer electronics due to its low power consumption and superior performance.

20

During operation of the non-volatile memory, free space may need to be created as new data is written and old data remains un-erased. Memory processes to create free space often requires relocating valid data into different parts of the memory array in order to more efficiently use the memory array for subsequent writing operations. Conventional systems maintain virtual blocks that map to physical blocks within the memory array, and such virtual blocks are often statically constructed within the firmware of the memory device each time that free space is needed. This static construction may lead to less efficient management of relocation of data when blocks are erased that have greater amounts of valid data than needed to be relocated in comparison with other blocks that may be linked to different virtual blocks.

25

30

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a schematic block diagram of a memory device according to embodiments of the present disclosure.

- 2 -

FIG. 2 shows a schematic block diagram of a software architecture for the memory controller of FIG. 1.

FIGS. 3A through 3E are simplified schematic block diagrams of the memory array of FIG. 1 for the purpose of illustrating garbage collection processes.

5 FIG. 4 is a flowchart illustrating a method for performing a garbage collection operation for a memory device according to an embodiment of the disclosure.

FIG. 5 is a schematic block diagram of an electronic system according to an embodiment of the present disclosure.

10 MODE(S) FOR CARRYING OUT THE INVENTION

The illustrations included herewith are not meant to be actual views of any particular systems, memory device, architecture, or process, but are merely idealized representations that are employed to describe embodiments herein. Elements and features common between figures may retain the same numerical designation except that, for ease of following the description, for the most part, reference numerals begin with the number of the drawing on which the elements are introduced or most fully described. In addition, the elements illustrated in the figures are schematic in nature, and many details regarding the physical layout and construction of a memory array and/or all steps necessary to access data may not be described as they would be understood by those of ordinary skill in the art.

20 Memory devices (e.g., including memory arrays divided into blocks) are described, as are related methods of operation and electronic systems. In some embodiments, a memory device comprises a memory array having non-volatile memory cells, and a memory controller operably coupled to the memory array. The memory controller is configured to dynamically construct a superblock during each garbage collection process based, at least in part, on an amount of valid data present in each physical block of the memory array. The memory controller may include a flash translation layer configured to perform the garbage collection and address mapping for the memory controller, and may manage dies across multiple channels (e.g., a first channel, a second channel, etc.). In some embodiments a superblock may be constructed by selecting a physical block from either the first die or the second die that has a lowest amount of valid data for the first channel, and selecting another physical block from either the third die or the fourth die that has a lowest amount of valid data for the second channel. In some embodiments, the superblock may be constructed by selecting multiple physical blocks one or more of the

- 3 -

first die or the second die, and selecting multiple physical blocks from one or more of the third die or the fourth die. The size of the superblock may also be dynamically selected for each garbage collection process based, at least in part, on a number of physical blocks needed to be cleared during a given garbage collection process. For example, a first
5 superblock may be constructed according to a first set of rules during a first garbage collection process, and a second superblock may be constructed according to a second set of rules during a second garbage collection process.

As used herein, the singular forms “a,” “an,” and “the” are intended to include the plural forms as well, unless the context clearly indicates otherwise.

10 As used herein, “and/or” includes any and all combinations of one or more of the associated listed items.

As used herein, the term “configured” refers to a size, shape, material composition, and arrangement of one or more of at least one structure and at least one apparatus facilitating operation of one or more of the structure and the apparatus in a pre-determined way.

15 As used herein, the phrase “coupled to” refers to structures operatively connected with each other, such as electrically connected through a direct ohmic connection or through an indirect connection (e.g., via another structure).

As used herein, “garbage collection” refers to a process performed on a memory device that includes the recovery of good data (i.e., valid data) from bad data (i.e., invalid data), and re-distribution of the valid data back into a pool of available physical blocks of
20 the memory array.

As used herein, a “superblock” is a virtual (i.e., logical) combination of physical blocks. The linking of a set of physical blocks as a superblock may be managed by the memory controller.

25 FIG. 1 is a schematic block diagram of a memory device 100 according to embodiments of the present disclosure. The memory device 100 includes a memory controller 110 operably coupled with a memory array 120. The memory array 120 may include non-volatile storage memory that may be written to, read from, and/or erased as controlled by the memory controller 110. The memory array 120 may be physically
30 constructed over one or more physical dies (e.g., multiple integrated circuits), and each physical die (also referred to as “die” or “semiconductor die”) may include a fixed number of physical data blocks (also referred to as “physical blocks” or “blocks”) that includes a fixed number of pages. The basic unit for a read/write operations is often a page. The basic unit for

- 4 -

erase operations is often a block. As an example, a die may include 4 blocks, each block may include 32 pages, and each page may include 512 bytes. Physical dies may also reserve space in the memory array 120 for replacement or spare area. Of course, any specific reference to a number of dies, block size, or other measure of capacity is to be understood as an example.

- 5 Embodiments of the present disclosure include memory arrays of a variety of different configurations and capacity. For example, memory cells of a block may be configured as single-level cells (SLC) and/or multi-level cells (MLC).

As non-limiting example, the memory array 120 may be configured as non-volatile memory, such as NAND Flash, NOR Flash, erasable programmable read only memory
10 (EPROM), electrically erasable programmable read only memory (EEPROM), ferroelectric RAM (FRAM), magnetoresistive RAM (MRAM) or other suitable non-volatile memory technology or any combination thereof. The memory device 100 may be incorporated within an embedded multi-media controller (also referred to as “embedded MMC” or “eMMC”), Solid State Drives (SSD), Universal Flash Storage (UFS) devices, and/or other
15 devices that utilize non-volatile memory as discussed herein.

The memory controller 110 may be configured to control operation of the memory array 120, including issuing the commands for reading, writing, and/or erasing. In addition, the memory controller 110 may be configured to maintain a logical mapping of the memory array 120 for performing the operations. The logical mapping may include page-level
20 mapping, block-level mapping, and/or hybrid-level mapping. During operation, the memory array 120 may monitor the status of data within the different physical blocks of the memory array 120 to determine if free space is needed and/or if valid data should be moved to different physical blocks improve the operational efficiency.

The memory controller 110 may be configured to perform a function that is generally
25 referred to in the art as “garbage collection.” As described above, garbage collection includes the separation and recovery of good data (i.e., valid data) from bad data (i.e., invalid data), and re-distribution of the valid data back into the pool of available physical blocks of the memory array 120. During garbage collection, the memory controller 110 may operate on more than one physical block at a time. This virtual (i.e., logical)
30 combination of physical blocks is referred to as a superblock. Conventionally, superblocks are static such that the same physical blocks are constructed (e.g., linked with each other) as the same superblock each time garbage collection occurs. Occasionally, new superblocks may be formed if a physical block become defective; however, embodiments

- 5 -

of the present disclosure include constructing a new superblock dynamically each time garbage collection occurs for the physical blocks regardless of whether any physical blocks are determined to be bad. Rather, the combination of physical blocks created to construct a superblock may be based, at least in part, on the amount of valid data in each of the
5 physical blocks. Additional details of this process will be discussed below with respect to FIGS. 3A-3E and FIG. 4.

FIG. 2 shows a schematic block diagram of a software architecture for the memory controller 110 of FIG. 1. The memory controller 110 may be configured to execute and manage one or more applications (via application layer 202), file systems (via file system
10 layer 204), and translation (via flash translation layer (FTL) 206) when communicating with the memory array 120. The application layer 202 provides high level services for application programs being run by the memory controller 110. The file system layer 204 issues a series of read or write commands each with a logical sector number, to read data from, or write data to, specific addresses in the memory array. The FTL 206 may also emulate the functionality of
15 the physical blocks of the memory array 120, hiding the presence of erase operation and erase-before-write characteristics from the memory array 120.

The FTL 206 may be configured to perform various functions, at least two of which include address translation and garbage collection. The address translation function includes the FTL 206 performing logical-to-physical address mapping (i.e., mapping of a
20 logical system level address to a physical memory address). For example, the FTL 206 may convert the logical sector number issued by the file system layer 204 to a real physical sector number of the memory array 120 (e.g., by a mapping algorithm in the FTL 206). This mapping may include the FTL 206 maintaining a mapping table of virtual addresses from upper software layers to correspond with the physical addresses on the memory array 120.
25 The FTL 206 may also maintain a virtual mapping of multiple physical blocks referred to herein as a "superblock." Thus, superblocks are physical blocks are the logically grouped together by the block management unit of the FTL 206.

The garbage collection function includes the separation and recovery of valid data (i.e., "good" data) from invalid data (i.e., "bad" data), and re-distribution of the valid data
30 into the pool of available physical blocks and erasing invalid data to free up space for additional writes to occur. Superblocks may be maintained for purposes of parallel reading/writing of data, as well as for garbage collection. As discussed herein, the superblocks used for purposes of reading/writing may be different than the superblocks

- 6 -

used for garbage collection. In particular, the superblocks used for garbage collection may be dynamically constructed each time garbage collection occurs regardless of the construction of the superblocks used during reading/writing. At each instance of garbage collection, the dynamic superblock may be constructed based on an amount of valid data in each physical block of the memory array 120 so that the number of internal data moves
5 may be reduced when data is relocated prior to being erased.

FIGS. 3A through 3E are simplified schematic block diagrams of the memory array 120 of FIG. 1 for the purpose of illustrating garbage collection processes performed by the FTL 206 (FIG. 2). For comparison of functionality, FIG. 3A shows a conventional
10 garbage collection process using static superblocks, and FIG. 3B shows a garbage collection process using dynamic superblocks according to an embodiment of the present disclosure.

FIGS. 3A through 3C show dies 301, 302, 303, 304 having a plurality of blocks (blocks 1, 2). The dies 301, 302, 303, 304 are managed by two channels (channels 1, 2). A
15 channel may be a hardware unit within the memory controller 110 that enables the memory controller 110 to communicate with a set of dies at the same time. For example, the first channel (channel 1) manages the first die (die 1) 301 and the second die (die 2) 302, while the second channel (channel 2) manages the third die (die 3) 303 and the fourth die (die 4) 304. Each die 301, 302, 303, 304 includes multiple blocks (blocks 1, 2) having data stored
20 in pages therein. Units of valid data are indicated within the blocks in parentheses (e.g., (8), (6), etc.). Two channels are described herein for simplicity; however, it should be recognized that any number of channels are contemplated as embodiments of the present disclosure. Likewise, it should be understood that two dies per channel, and two blocks per die is described herein for simplicity, but that any number of dies and blocks are
25 contemplated as embodiments of the present disclosure.

Referring specifically to FIG. 3A, the FTL 206 may maintain static superblocks across the different channels. Each superblock may have at least one physical block logically linked to another physical block from another die of a different channel. For example, the first static superblock (static superblock 1) may logically link the first block
30 from the first die 301 and the first block of the third die 303, the second static superblock (static superblock 2) may logically link the second block from the first die 301 and the second block of the third die 303, the third static superblock (static superblock 3) may logically link the first block from the second die 302 and the first block of the fourth

- 7 -

die 304, and the fourth static superblock (static superblock 4) may logically link the second block from the second die 302 and the second block of the fourth die 304. In some situations, superblocks may be redefined if a physical block becomes defective and is no longer used by the memory array 120. However, so long as the physical blocks remain
5 usable, these superblocks will be defined as such whenever garbage collection occurs due to their static nature. In addition, these superblocks may also be the same superblocks used during parallel read and write operations.

During garbage collection, the FTL 206 may select a static superblock to clean up its physical blocks. In some conventional methods, the FTL 206 may select the static
10 superblock with the least amount of valid data combined among its physical blocks in order to clear the most blocks while moving the least amount of valid data. In the example shown in FIG. 3B, the first static superblock has 24 units of valid data, the second static superblock has 18 units of valid data, the third static superblock has 12 units of valid data, the fourth static superblock has 20 units of valid data. Thus, in some conventional
15 methods, the FTL 206 would select the third static superblock during garbage collection to clean up the first block of the second die and the first block of the fourth die. The FTL 206 would then move the 12 valid data units to another physical block, and then erase the data from its physical blocks. If that process cleared enough free space, the garbage collection process would end; however, if more free space is needed, the FTL 206 would select
20 another superblock (e.g., static superblock 2) to move its valid data to another physical block and erase the data from its physical blocks.

Referring specifically to FIG. 3B, the FTL 206 may construct dynamic superblocks across the different channels each time garbage collection occurs. Each superblock may have at least one physical block logically linked to another physical block from another die
25 of a different channel. During garbage collection, the FTL 206 may construct a dynamic superblock to clean up a set of physical blocks. The FTL 206 may construct each new dynamic superblock by selecting a physical block from the first channel, another physical block from the second channel, as well as physical blocks from additional channels (if any).

30 The FTL 206 may construct each new dynamic superblock based on the amount of valid data in each physical block of each die. For example, the physical block from each channel may be selected that has the least amount of valid data. In the example shown in FIG. 3B, the FTL 206 may search the dies 301, 302 of the first channel and select block 2

- 8 -

of the first die 301 because (6) is the lowest valid data for physical blocks across the first channel. The FTL 206 may also search the dies 303, 304 of the second channel and select block 1 of the fourth die 304 because (2) is the lowest valid data for physical blocks across the second channel. The FTL 206 would then move the 8 valid data units to another physical block, and then erase physical blocks used in the dynamic superblock. If that process cleared enough free space, the garbage collection process may end; however, if more free space is needed, the FTL 206 may construct another dynamic superblock (e.g., block 1 of the first die 301 and block 2 of the fourth die 304) to move its valid data to another physical block and erase the data from its physical blocks. As a result of the garbage collection process of FIG. 3B, only 8 valid data units were needed to be moved to clear a superblock in contrast to the 12 valid data units for the garbage collection process of FIG. 3A. Thus, the amount of data needed to be relocated may be reduced during garbage collection in comparison to conventional methods.

Referring specifically to FIG. 3C, the FTL 206 may construct a new dynamic superblock by selecting a physical block from each physical die for each channel for a garbage collection process. For example, the new dynamic superblock may include a physical block from the first die 301, a physical block from the second die 302, a physical block from the third die 303, a physical block from the fourth die 304, as well as physical blocks from additional dies (if any). The FTL 206 may construct each new dynamic superblock based on the amount of valid data in each physical block of each die. For example, the physical block from each die may be selected that has the least amount of valid data. In the example shown in FIG. 3C, the FTL 206 may search the dies 301, 302, 303, 304 and select block 2 of the first die 301 because (6) is the lowest valid data for the first die 301, block 1 of the second die 302 because (10) is the lowest valid data for the second die 302, block 2 of the third die 303 because (12) is the lowest valid data for the third die 303, and block 2 of the fourth die 304 because (2) is the lowest valid data for the fourth die 304.

Referring specifically to FIG. 3D, in some embodiments the FTL 206 may construct a new dynamic superblock by selecting multiple physical blocks from the same physical die for a garbage collection process. For example, the new dynamic superblock may include two physical blocks from the first die 301, and two physical blocks from the fourth die 304. As with other embodiments, the FTL 206 may construct the new dynamic superblock based on the amount of valid data in each physical block of each die. For

- 9 -

example, the physical block from each die may be selected that has the least amount of valid data. In the example shown in FIG. 3D, the FTL 206 may search the dies 301, 302 and select block 4 of the first die 301 because (4) is the lowest valid data for the dies 301, 302 of the first channel, and block 2 of the first die 301 because (6) is the next lowest valid data for the dies 301, 302 of the first channel. In addition, the FTL 206 may search the dies 303, 304 and select block 1 of the fourth die 304 because (2) is the lowest valid data for the dies 303, 304 of the second channel, and block 4 of the fourth die 304 because (4) is the next lowest valid data for the dies 303, 304 of the second channel.

Additional combinations of blocks and dies are also contemplated as embodiments of the disclosure, which may depend, in part, on the size of the dynamically constructed superblock. For example, constructing a dynamic superblock to include six physical blocks may cause the FTL 206 to add block 3 of the second die 302 and block 3 of the third die 303 as they have the next lowest valid data for each respective channel (see, e.g., FIG. 3E). In some embodiments, the FTL 206 may be configured to construct a superblock that includes physical blocks from dies within a single channel only. In other words, when searching the dies 301, 302, 303, 304, the FTL 206 may not be bound to select blocks from both channels. As a result, a dynamic superblock may be constructed by linking only physical blocks from dies in channel 2 if it so happens that channel 2 includes the blocks with the four lowest valid data for a dynamic superblock having a size of four physical blocks.

FIG. 4 is a flowchart 400 illustrating a method for performing a garbage collection operation for a memory device according to an embodiment of the disclosure. At operation 410, the memory controller (e.g., via firmware for the FTL) may determine that cleanup is desirable to free space in some of the physical blocks of the memory array. Garbage collection may be initiated, and a dynamic superblock may be constructed each time garbage collection is performed.

At operation 420, the memory controller (e.g., via firmware for the FTL) may search through the physical blocks managed by a first channel to determine the amount of valid data for each physical block of each die managed by the first channel.

At operation 430, the memory controller (e.g., via firmware for the FTL) may search through the physical blocks supported by a first channel to determine the amount of valid data for each physical block of each die supported by the first channel. If there are additional channels, additional blocks may be similarly searched.

- 10 -

At operation 440, the dynamic superblock may be constructed including a set of physical blocks that are linked together based, at least in part, on the amount of valid data in each of the physical blocks. In some embodiments, constructing the dynamic superblock includes linking at least one physical block for the first channel with at least one physical
5 block for the second channel. As an example, the dynamic superblock may be constructed with the blocks selected from first channel (operation 420) and/or the second channel (operation 430) and any additional channels. In some embodiments, the physical block from the first channel with the minimum amount of valid data may be selected, and the physical block from the second channel with the minimum amount of valid data may be
10 selected. In some embodiments, multiple physical blocks from the same channel may be selected. The multiple physical blocks from the same channel may or may not on the same die in different embodiments.

At operation 450, valid data may be moved (i.e., relocated) from physical blocks defined by the superblock to available space within another physical block in the memory
15 array. At operation 460, the data in the physical blocks defined by the superblock may be erased.

After the physical blocks are erased at operation 460, then the firmware can write new data to the newly available space. The firmware may continue to read, write, and/or erase data according to the operation of the application and file system layers until the FTL determines
20 that additional free space is needed and garbage collection is needed. A new dynamic superblock may then be constructed accordingly, which may include different physical blocks depending on the valid data contained within each physical block for die continues and then would do the same operation the next time that cleanup is needed.

In some embodiments, the size of the new dynamic superblock may be determined at
25 the outset (e.g., operation 410) when it is determined that free space is needed. For example, it may be determined that a superblock having four physical blocks is needed for a first garbage collection process. As a result, rules for constructing a superblock with four physical blocks may be employed. In another instance, it may be determined that a superblock having six physical blocks is needed for a second garbage collection process. As a result, rules for
30 constructing a superblock with six physical blocks may be employed. Thus, different sets of rules for constructing different superblocks may be employed depending on the size of superblock needed for a given garbage collection operation.

- 11 -

FIG. 5 is a schematic block diagram of an electronic system 500 according to an embodiment of the present disclosure. The electronic system 500 includes a processor 510 operably coupled with a memory device 520, one or more input devices 530 and one or more output devices 540. The electronic system 500 may be a consumer electronic device, such as a desktop computer, a laptop computer, a tablet computer, an electronic reader, a smart phone or other type of communication device, as well as any type of computing system incorporating non-volatile storage.

As discussed above, the memory device 520 may be configured to dynamically generate superblocks for a garbage collection process as discussed above. The memory controller (including the FTL) may be integrated with the memory device 520 (e.g., the same chip or circuit board as the physical dies for the memory array). In some embodiments, the memory controller (including the FTL) may be integrated with the processor 510 that is configured to control other features (e.g., run the operating system) of the electronic system 500.

Additional non-limiting embodiments include:

Embodiment 1. A memory device, comprising: a memory array having non-volatile memory cells; and a memory controller operably coupled to the memory array, the memory controller configured to dynamically construct a superblock for at least one garbage collection process based, at least in part, on an amount of valid data present in each physical block of the memory array.

Embodiment 2. The memory device of Embodiment 1, wherein the memory controller includes a flash translation layer configured to perform the garbage collection and address mapping for the memory controller.

Embodiment 3. The memory device of Embodiment 1 or Embodiment 2, wherein the memory array includes dies having different physical blocks of data.

Embodiment 4. The memory device of Embodiment 3, wherein the dies include: a first die and a second die managed by a first channel of the memory controller; and a third die and a fourth die managed by a second channel of the memory controller.

Embodiment 5. The memory device of Embodiment 4, wherein the memory controller is configured to construct the superblock by selecting a physical block from either the first die or the second die that has a lowest amount of valid data for the first channel, and selecting another physical block from either the third die or the fourth die that has a lowest amount of valid data for the second channel.

- 12 -

Embodiment 6. The memory device of Embodiment 4, wherein the memory controller is configured to construct the superblock by selecting multiple physical blocks one or more of the first die or the second die, and selecting multiple physical blocks from one or more of the third die or the fourth die.

5 Embodiment 7. The memory device of any of Embodiments 1 through 6, wherein the memory controller is further configured to dynamically determine a size for the superblock for the at least one garbage collection process based, at least in part, on a number of physical blocks needed to be cleared during a given garbage collection process.

Embodiment 8. The memory device of Embodiment 7, wherein the memory
10 controller is configured to construct a first superblock according to a first set of rules during a first garbage collection process, and a second superblock according to a second set of rules during a second garbage collection process.

Embodiment 9. The memory device of any of Embodiments 1 through 8, wherein
15 the non-volatile memory cells of the memory array are configured as at least one of NAND Flash, NOR Flash, erasable programmable read only memory (EPROM), electrically erasable programmable read only memory (EEPROM), ferroelectric RAM (FRAM), or magnetoresistive RAM (MRAM).

Embodiment 10. A memory device, comprising: physical blocks of memory cells;
20 and a memory controller configured to construct a new superblock dynamically each time garbage collection occurs for the physical blocks regardless of whether any physical blocks are determined to be bad.

Embodiment 11. The memory device of Embodiment 10, wherein the memory controller is configured to execute firmware having an application layer, a file system layer, and a translation layer.

25 Embodiment 12. The memory device of Embodiment 11, wherein the translation layer includes a Flash Translation Layer (FTL) configured to construct the new superblock and perform the garbage collection.

Embodiment 13. The memory device of Embodiment 12, wherein the FTL is
30 configured to construct the new superblock responsive to determining a set of physical blocks having a minimum total amount of valid data.

Embodiment 14. The memory device of Embodiment 13, wherein the set of physical blocks includes at least one physical block from a die that is part of a channel

- 13 -

managed by the memory controller, and at least one additional physical block from an additional die that is part of another channel managed by the memory controller.

Embodiment 15. The memory device of Embodiment 13 or Embodiment 14, wherein the set of physical blocks includes physical blocks from different dies that
5 comprise the memory cells.

Embodiment 16. The memory device of any of Embodiments 13 through 15, wherein the set of physical blocks includes physical blocks from a common die that comprise the memory cells.

Embodiment 17. A method for managing operation of a memory device, the
10 method comprising: initiating a garbage collection process for a memory array divided into physical blocks of data; determining an amount of valid data for the physical blocks of data; and constructing a dynamic superblock based, at least in part, on the amount of valid data for the physical blocks of data for the memory array.

Embodiment 18. The method of Embodiment 17, further comprising: moving valid
15 data for physical blocks selected for construction of the dynamic superblock; and erasing the physical blocks selected for construction of the dynamic superblock.

Embodiment 19. The method of Embodiment 17 or Embodiment 18, wherein
initiating the garbage collection process includes: separating and recovering valid data from invalid data for the physical blocks of the dynamic superblock; re-distributing the
20 valid data into a pool of available physical blocks outside of the dynamic superblock; and erasing the physical blocks of the dynamic superblock to free up space therein for additional writes to occur.

Embodiment 20. The method of any of Embodiments 17 through 19, wherein
determining the amount of valid data for the physical blocks of data includes: searching
25 through physical blocks for each die managed by a first channel to determine valid data for each physical block of the first channel; and searching through physical blocks for each die managed by a second channel to determine valid data for each physical block of the second channel.

Embodiment 21. The method of Embodiment 20, wherein constructing the
30 dynamic superblock includes linking at least one physical block for the first channel with at least one physical block for the second channel.

Embodiment 22. The method of Embodiment 21, wherein linking at least one physical block for the first channel with at least one physical block for the second channel

- 14 -

includes: selecting the at least one physical block for the first channel having a minimum amount of valid data among the physical blocks for the first channel; and selecting the at least one physical block for the second channel having a minimum amount of valid data among the physical blocks for the second channel.

5 Embodiment 23. An electronic system, comprising: a memory device including dies having physical blocks, the memory device configured to link a first set of physical blocks to dynamically construct a first superblock responsive to a first garbage collection process, and link a second set of physical blocks to dynamically construct a second superblock responsive to a second garbage collection process.

10 Embodiment 24. The electronic system of Embodiment 23, further comprising at least one of an embedded multi-media controller (eMMC), a Solid State Drive (SSD), or a Universal Flash Storage (UFS) devices that includes the memory device.

 Embodiment 25. The electronic system of Embodiment 23 or Embodiment 24,
15 further comprising at least one processor, at least one input device, and at least one output device.

 While the disclosure is susceptible to various modifications and alternative forms, specific embodiments have been shown by way of example in the drawings and have been described in detail herein. However, the disclosure is not limited to the particular forms disclosed. Rather, the disclosure is to cover all modifications, equivalents, and alternatives
20 falling within the scope of the following appended claims and their legal equivalents.

- 15 -

CLAIMS

What is claimed is:

- 5 1. A memory device, comprising:
a memory array having non-volatile memory cells; and
a memory controller operably coupled to the memory array, the memory controller
 configured to dynamically construct a superblock for at least one garbage collection
 process based, at least in part, on an amount of valid data present in each physical
10 block of the memory array.
2. The memory device of claim 1, wherein the memory controller includes a
flash translation layer configured to perform the garbage collection and address mapping
for the memory controller.
- 15 3. The memory device of claim 1 or claim 2, wherein the memory array
includes dies having different physical blocks of data.
4. The memory device of claim 3, wherein the dies include:
20 a first die and a second die managed by a first channel of the memory controller; and
a third die and a fourth die managed by a second channel of the memory controller.
5. The memory device of claim 4, wherein the memory controller is configured
to construct the superblock by selecting a physical block from either the first die or the
25 second die that has a lowest amount of valid data for the first channel, and selecting
another physical block from either the third die or the fourth die that has a lowest amount
of valid data for the second channel.
6. The memory device of claim 4, wherein the memory controller is configured
30 to construct the superblock by selecting multiple physical blocks one or more of the first
die or the second die, and selecting multiple physical blocks from one or more of the third
die or the fourth die.

- 16 -

7. The memory device of claim 1 or claim 2, wherein the memory controller is further configured to dynamically determine a size for the superblock for the at least one garbage collection process based, at least in part, on a number of physical blocks needed to be cleared during a given garbage collection process.

5

8. The memory device of claim 7, wherein the memory controller is configured to construct a first superblock according to a first set of rules during a first garbage collection process, and a second superblock according to a second set of rules during a second garbage collection process.

10

9. The memory device of claim 1 or claim 2, wherein the non-volatile memory cells of the memory array are configured as at least one of NAND Flash, NOR Flash, erasable programmable read only memory (EPROM), electrically erasable programmable read only memory (EEPROM), ferroelectric RAM (FRAM), or magnetoresistive RAM (MRAM).

15

10. A memory device, comprising:
physical blocks of memory cells; and
a memory controller configured to construct a new superblock dynamically each time
garbage collection occurs for the physical blocks regardless of whether any physical
blocks are determined to be bad.

20

11. The memory device of claim 10, wherein the memory controller is configured to execute firmware having an application layer, a file system layer, and a translation layer.

25

12. The memory device of claim 11, wherein the translation layer includes a Flash Translation Layer (FTL) configured to construct the new superblock and perform the garbage collection.

30

13. The memory device of claim 12, wherein the FTL is configured to construct the new superblock responsive to determining a set of physical blocks having a minimum total amount of valid data.

- 17 -

14. The memory device of claim 13, wherein the set of physical blocks includes at least one physical block from a die that is part of a channel managed by the memory controller, and at least one additional physical block from an additional die that is part of another channel managed by the memory controller.

15. The memory device of claim 13 or claim 14, wherein the set of physical blocks includes physical blocks from different dies that comprise the memory cells.

16. The memory device of claim 13 or claim 14, wherein the set of physical blocks includes physical blocks from a common die that comprise the memory cells.

17. A method for managing operation of a memory device, the method comprising:
initiating a garbage collection process for a memory array divided into physical blocks of data;
determining an amount of valid data for the physical blocks of data; and
constructing a dynamic superblock based, at least in part, on the amount of valid data for the physical blocks of data for the memory array.

18. The method of claim 17, further comprising:
moving valid data for physical blocks selected for construction of the dynamic superblock;
and
erasing the physical blocks selected for construction of the dynamic superblock.

19. The method of claim 17 or claim 18, wherein initiating the garbage collection process includes:
separating and recovering valid data from invalid data for the physical blocks of the dynamic superblock;
re-distributing the valid data into a pool of available physical blocks outside of the dynamic superblock; and
erasing the physical blocks of the dynamic superblock to free up space therein for additional writes to occur.

- 18 -

20. The method of claim 17 or claim 18, wherein determining the amount of valid data for the physical blocks of data includes:

5 searching through physical blocks for each die managed by a first channel to determine

valid data for each physical block of the first channel; and

searching through physical blocks for each die managed by a second channel to determine valid data for each physical block of the second channel.

21. The method of claim 20, wherein constructing the dynamic superblock
10 includes linking at least one physical block for the first channel with at least one physical block for the second channel.

22. The method of claim 21, wherein linking at least one physical block for the first channel with at least one physical block for the second channel includes:

15 selecting the at least one physical block for the first channel having a minimum amount of valid data among the physical blocks for the first channel; and

selecting the at least one physical block for the second channel having a minimum amount of valid data among the physical blocks for the second channel.

20 23. An electronic system, comprising:
a memory device including dies having physical blocks, the memory device configured to link a first set of physical blocks to dynamically construct a first superblock responsive to a first garbage collection process, and link a second set of physical blocks to dynamically construct a second superblock responsive to a second
25 garbage collection process.

24. The electronic system of claim 23, further comprising at least one of an embedded multi-media controller (eMMC), a Solid State Drive (SSD), or a Universal Flash Storage (UFS) devices that includes the memory device.

30

25. The electronic system of claim 23 or claim 23, further comprising at least one processor, at least one input device, and at least one output device.

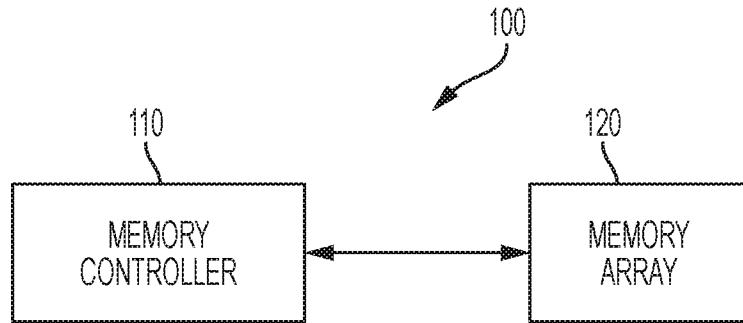


FIG. 1

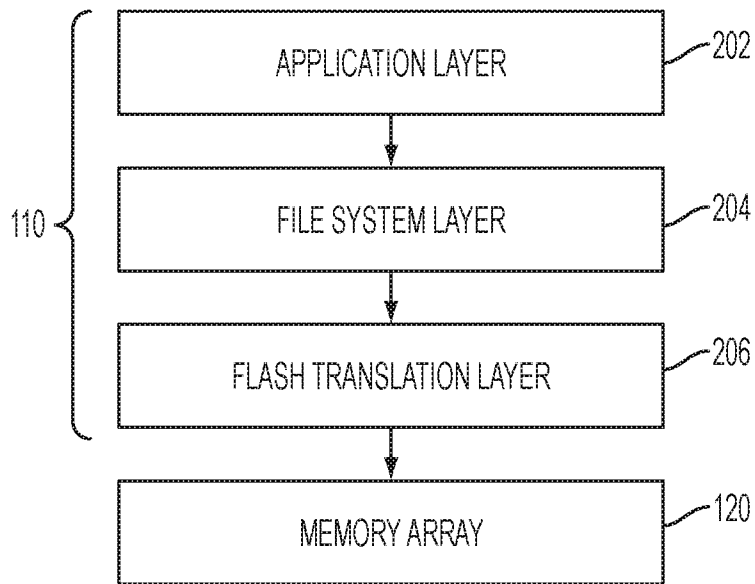


FIG. 2

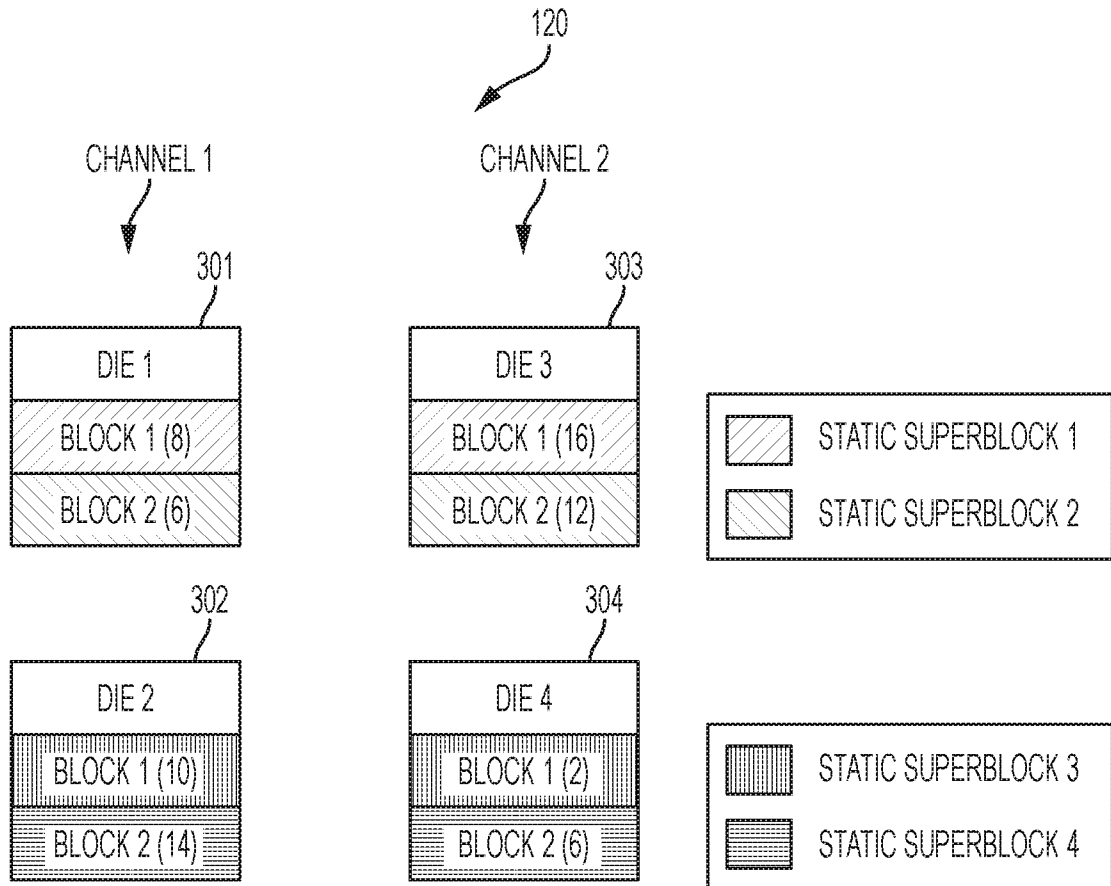


FIG. 3A
PRIOR ART

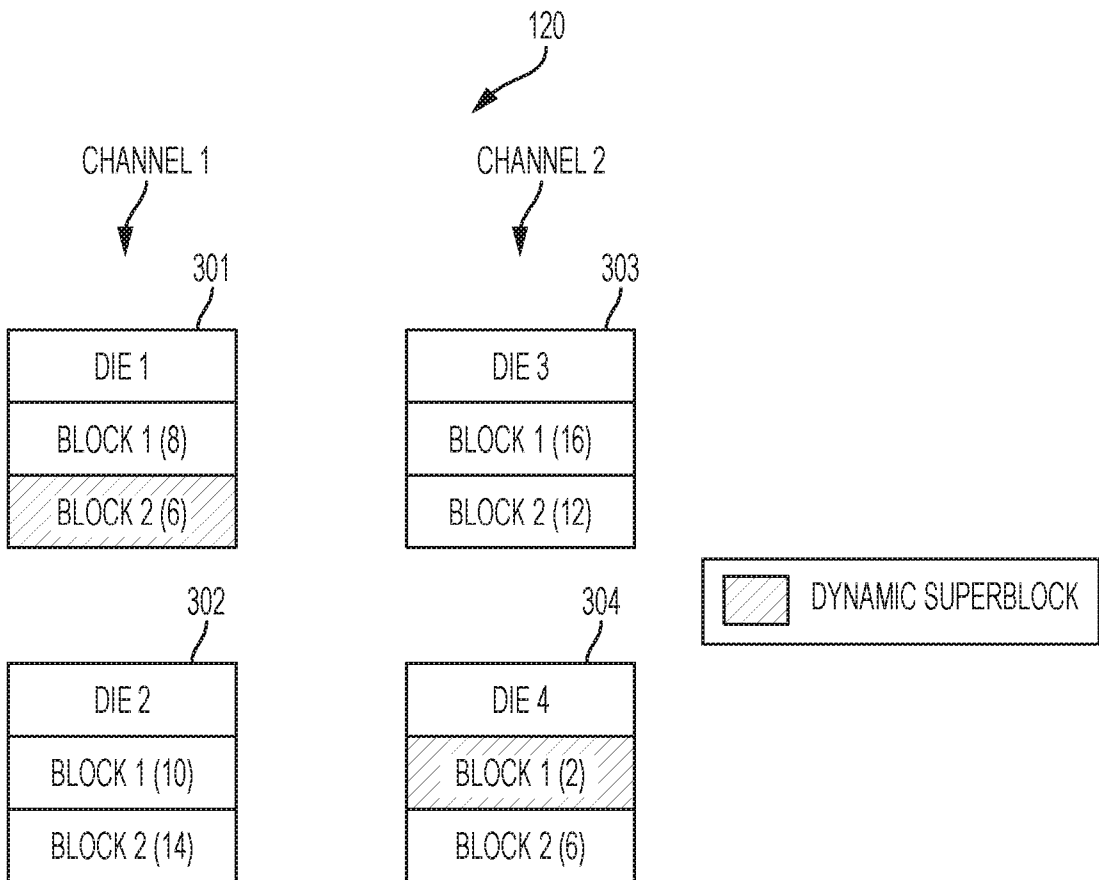


FIG. 3B

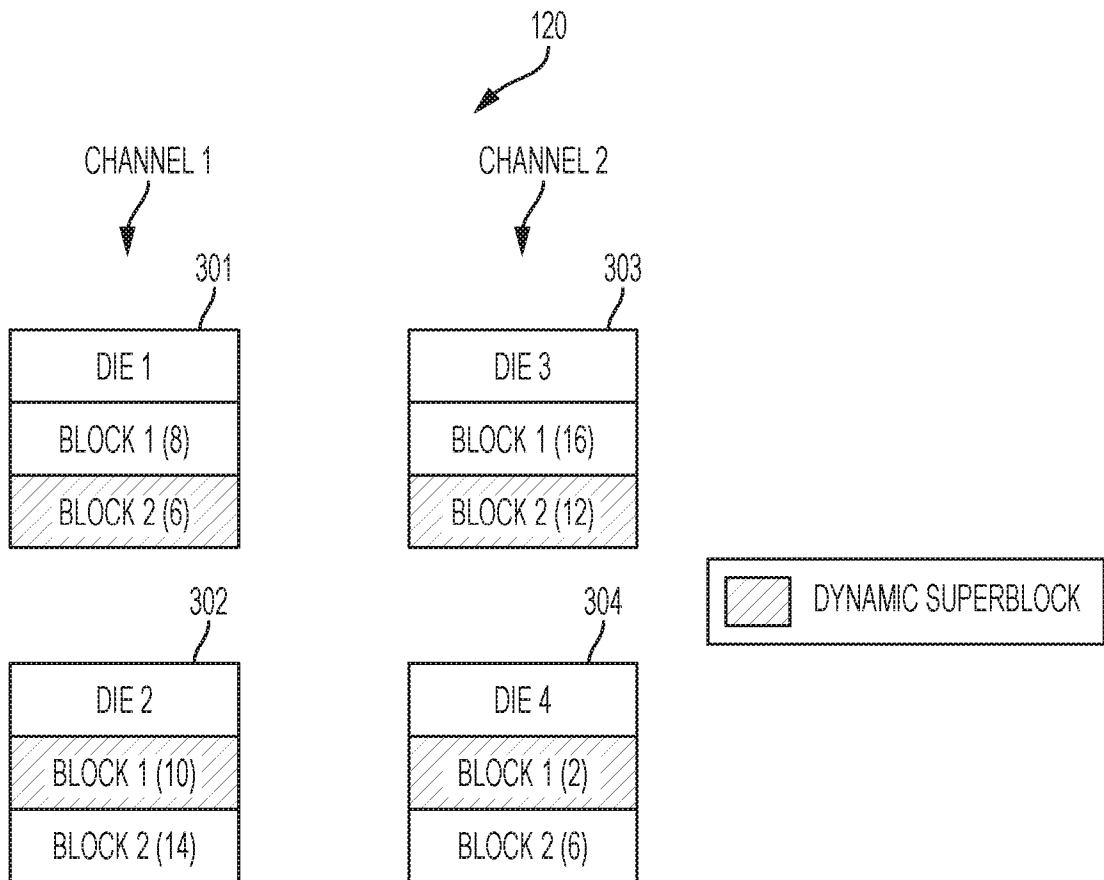


FIG. 3C

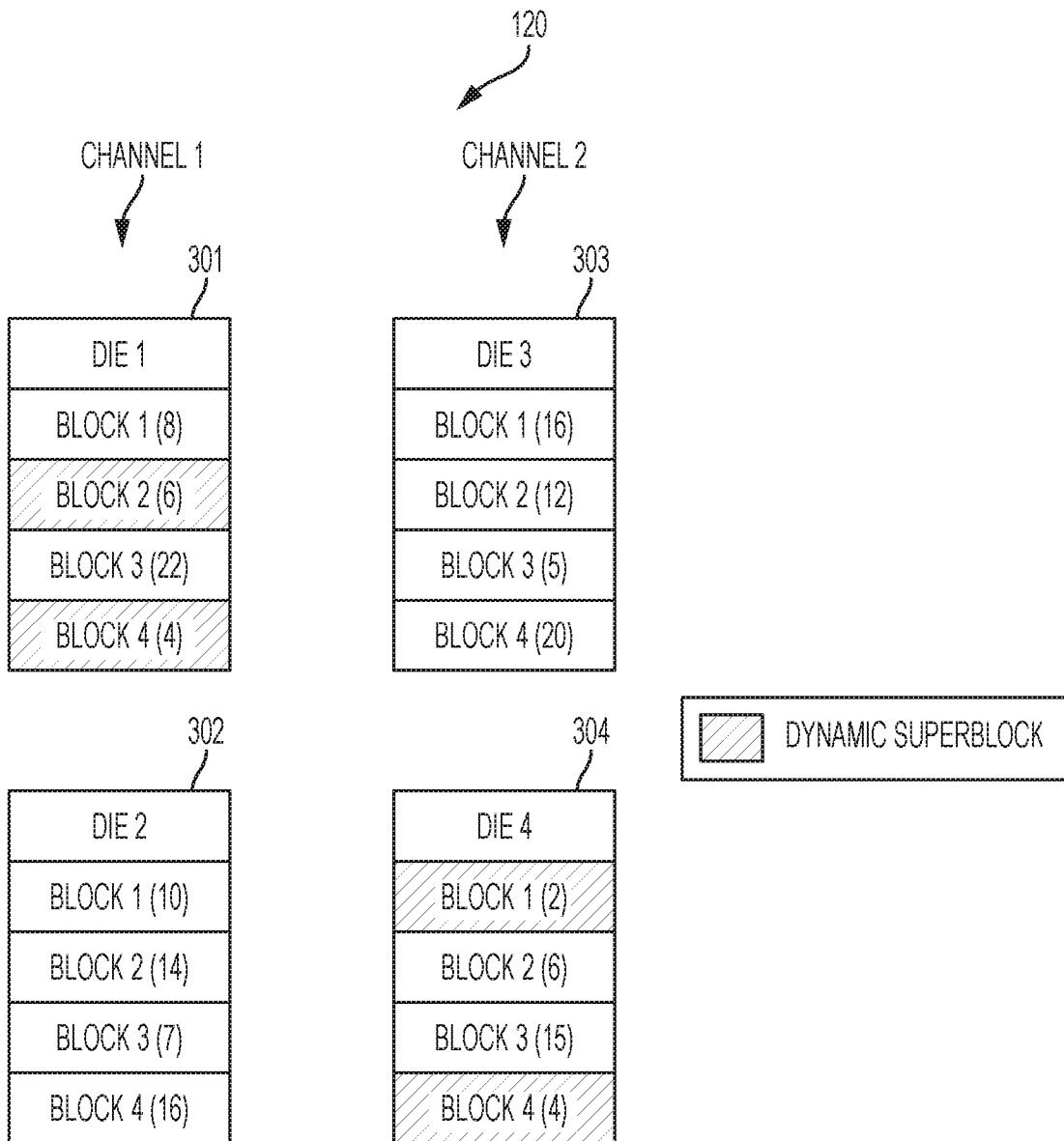


FIG. 3D

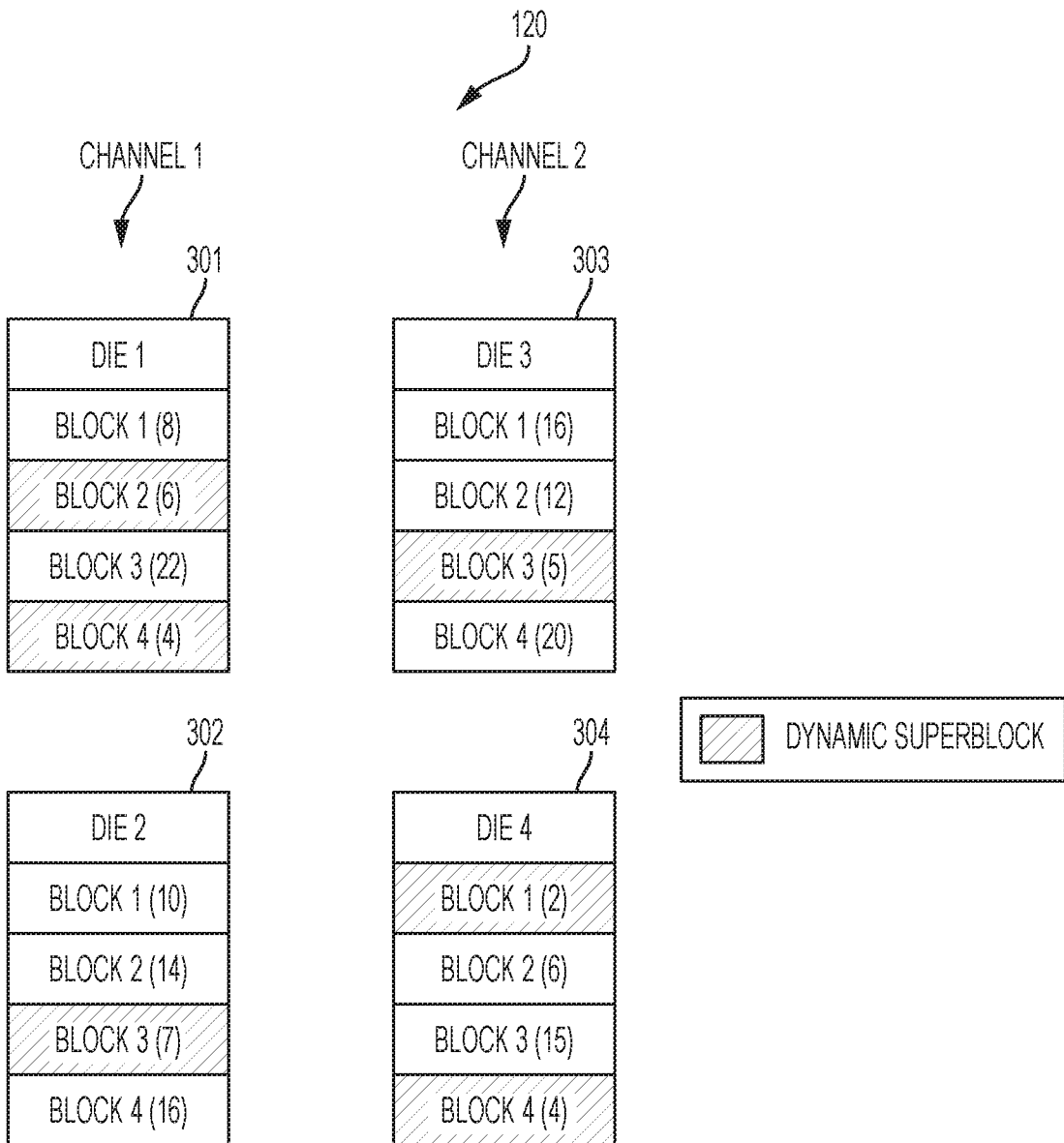


FIG. 3E

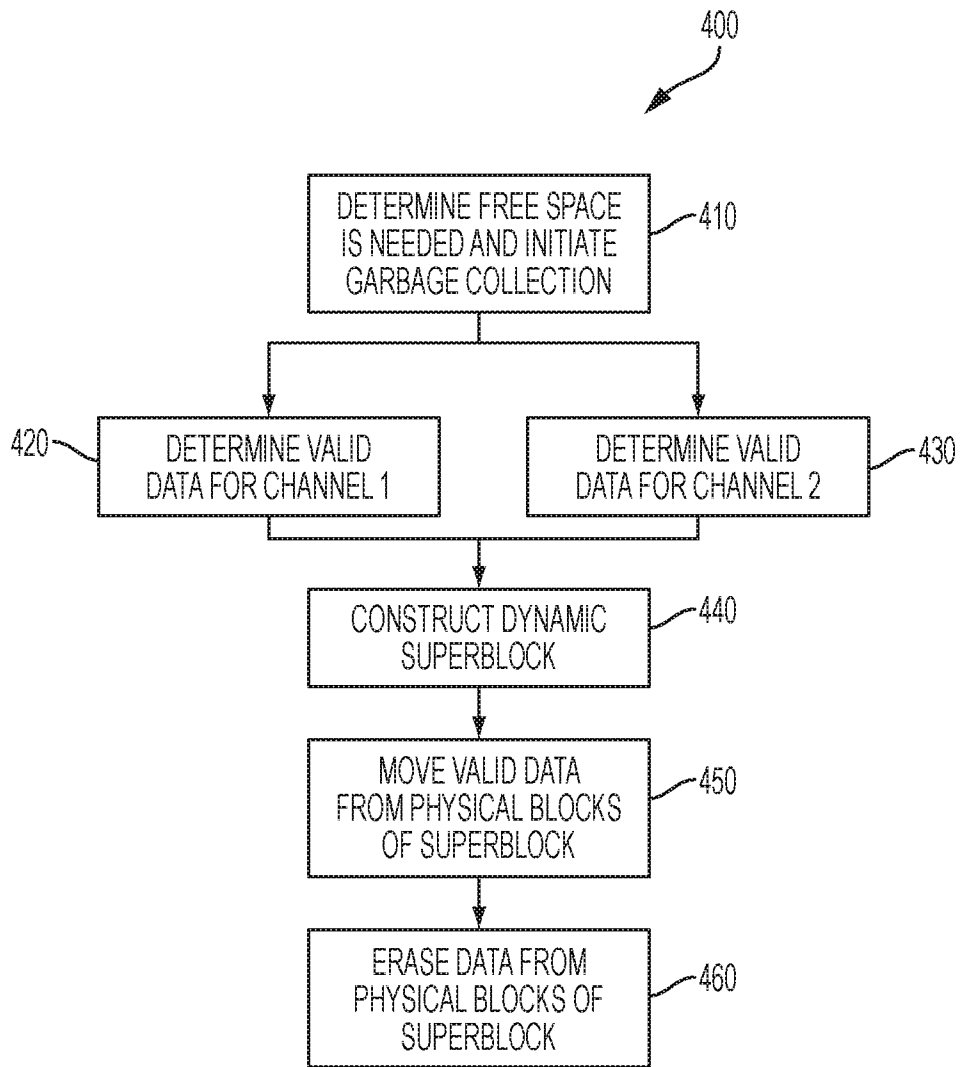


FIG. 4

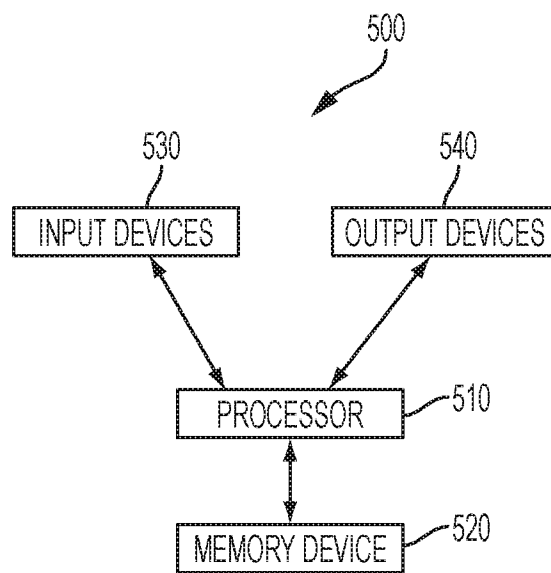


FIG. 5

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2017/021071**A. CLASSIFICATION OF SUBJECT MATTER****G06F 12/02(2006.01)i, G06F 13/16(2006.01)i, G06F 3/06(2006.01)i**

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F 12/02; G06F 11/26; G06F 3/06; G06F 12/10; G06F 11/10; G06F 17/30; G06F 12/00; G06F 13/16

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

Korean utility models and applications for utility models

Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

eKOMPASS(KIPO internal) & Keywords: memory, dynamic, superbloc, garbage, collection, die, select

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2012-0047409 A1 (DANIEL J. POST et al.) 23 February 2012 See paragraphs [0005]-[0008], [0019]-[0022], [0033], [0041], [0049], [0052], [0059], [0083], [0088], [0097]-[0099]; claim 9; and figure 1.	1-25
Y	US 2011-0283049 A1 (HO-FAN KANG et al.) 17 November 2011 See paragraphs [0009]-[0011], [0015]-[0016]; claim 1; and figure 1.	1-25
A	US 2015-0193300 A1 (APPLE INC.) 09 July 2015 See paragraphs [0042], [0087]; and figure 1.	1-25
A	US 2015-0317326 A1 (VMWARE, INC.) 05 November 2015 See paragraphs [0003], [0011]; and figure 1.	1-25
A	WO 2007-081598 A2 (SANDISK CORPORATION et al.) 19 July 2007 See paragraphs [0008], [0119]; and figure 10.	1-25

 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

12 June 2017 (12.06.2017)

Date of mailing of the international search report

12 June 2017 (12.06.2017)

Name and mailing address of the ISA/KR

International Application Division

Korean Intellectual Property Office

189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea



Facsimile No. +82-42-481-8578

Authorized officer

CHIN, Sang Bum

Telephone No. +82-42-481-8398



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2017/021071

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2012-0047409 A1	23/02/2012	US 8832507 B2	09/09/2014
US 2011-0283049 A1	17/11/2011	CN 102243613 A CN 102243613 B HK 1160526 A1 US 9026716 B2	16/11/2011 25/11/2015 08/07/2016 05/05/2015
US 2015-0193300 A1	09/07/2015	US 2014-157083 A1 US 2014-157085 A1 US 2016-299814 A1 US 8572311 B1 US 8677203 B1 US 8954823 B2 US 9015559 B2 US 9400710 B2	05/06/2014 05/06/2014 13/10/2016 29/10/2013 18/03/2014 10/02/2015 21/04/2015 26/07/2016
US 2015-0317326 A1	05/11/2015	None	
WO 2007-081598 A2	19/07/2007	TW 200733123 A TW I428920 B US 2007-0101095 A1 US 2007-0101096 A1 US 7509471 B2 US 7631162 B2 WO 2007-081598 A3	01/09/2007 01/03/2014 03/05/2007 03/05/2007 24/03/2009 08/12/2009 13/03/2008