STORAGE DEVICE AND STORING METHOD

CROSS-REFERENCE TO RELATED APPLICATION

This application is based upon and claims the benefit of priority from Japanese Patent Application No. 2015-194056; filed on September 30, 2015; the entire contents of which are incorporated herein by reference.

FIELD

Embodiments described herein relate generally to a storage device and a storing method.

BACKGROUND

In recent years, a storage device including non-volatile storage media of multiple types (for example, two types) whose access speeds and storage capacities are different from each other has been developed. As a representative of such storage devices, a hybrid driver is known. The hybrid driver includes in general a first non-volatile storage medium, and a second non-volatile storage medium with a low access speed and a large storage capacity, compared to the first non-volatile storage medium.

An example of related art includes JP-A-2009-129026.

DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram illustrating a configuration of a storage device according to an embodiment.

FIG. 2 is a conceptual diagram illustrating an example of a format of a storage area of a NAND memory according to the embodiment.

FIG. 3 is a flowchart illustrating an operation of read patrol with respect to the NAND memory according to the embodiment.

DETAILED DESCRIPTION

[0004]An exemplary embodiment provides a storage device whose reliability can be increased.

[0005]In general, according to one embodiment, a storage device includes a first non-volatile storage medium, a second non-volatile storage medium which has a plurality of blocks and can perform data processing at a faster speed than the first non-volatile storage medium, and a control unit which includes a first controller for controlling the first non-volatile storage medium, and a second controller for controlling the second non-volatile storage medium, in which the control unit acquires an error amount of first data which is read from the second non-volatile storage medium, and writes the read first data onto one of the first non-volatile storage medium and the second non-volatile storage medium in accordance with the error amount.

[0007]Hereinafter, an embodiment will be described with reference to the accompanying drawings.

[0008]In the present specification, an example in which multiple expressions are used for several elements is provided. Such expressions are just an example, and it does not deny that the elements are expressed in other expressions. In addition, elements to which multiple expressions are not used may also be expressed in different expressions.

[0009]In addition, the drawings are just schematic, and a relationship between a thickness and a planar dimension, a ratio between thickness of each layer, or the like may be different from actuality. In addition, portions in which relationships between dimensions or ratios between dimensions are different from each other may be included in the drawings.

[0010]FIG. 1 is a block diagram illustrating a configuration of a storage device 1 according to the present embodiment. The storage device 1 according to the present embodiment is, for example, a hybrid driver. The hybrid driver includes non-volatile storage media (that is, first non-volatile storage medium and a second non-volatile storage medium) of multiple types, for example, two types, in which access speeds and storage capacities are different from each other. In the present embodiment, the storage device 1 is described as a hybrid driver 1.

[0011]In the embodiment, a magnetic disk medium (hereinafter, referred to as a disk) 21 is used for the first non-volatile storage medium, and a NAND flash memory (hereinafter, referred to as a NAND memory) 11 is used for the second non-volatile storage medium. The disk 21 includes a system area (SA) 101 for recording management information, as will be described later. Access speed and storage capacity of the disk 21 are respectively slow and large, compared to those of the NAND flash memory 11.

[0012]The hybrid driver 1 illustrated in FIG. 1 is configured with a semiconductor drive unit 10 such as a solid state driver, and a hard disk drive unit (hereinafter, referred to as a HDD) 20. The semiconductor drive unit 10 includes the NAND memory 11 and a main controller (control unit) 27.

[0013]The NAND memory 11 in the hybrid driver 1 is used for various purposes. The NAND memory 11 is used, for example, performance improvement of the hybrid driver 1, a stable write operation when the hybrid driver 1 vibrates, fast start-up speed of the hybrid driver 1, or the like. As will be described later, the NAND memory 11 includes a system area (SA) 111 for recording management information.

[0014]The main controller 27 controls accessing to the NAND memory 11 in accordance with an access requirement (for example, a write requirement or a read requirement) from a host device (hereinafter, referred to as a host). In the present embodiment, in order for fast accessing to the hybrid driver 1 from the host, the NAND memory 11 is used as a cache (cache memory) for storing data which is recently accessed by the host. The host uses the hybrid driver 1 illustrated in FIG. 1 as a storage device therefor.

[0015]The main controller 27 is realized by, for example, a large scale integrated circuit (LSI) in which multiple elements are integrated in a single chip. The main controller 27 includes a memory interface controller (hereinafter, referred to as a memory IF) 122, a microprocessor unit (MPU) 123, a read only memory (ROM) 124, a random access memory (RAM) 125, a read and write (R/W channel) 271, and a hard disk controller (HDC) 272.

[0016]The memory IF (first interface controller) 122 is coupled to the NAND memory 11, and accesses the NAND memory 11 under a control of the MPU 123.

[0017]The MPU 123 performs processing (for example, write processing or read processing) for accessing the NAND memory 11 based on a command which is transferred from the main controller 27, in accordance with a first control programs. In the present embodiment, the first control program is stored in advance in, for example, the ROM 124.

[0018]Instead of the ROM 124, a rewritable non-volatile ROM, for example, a flash ROM may be used. A part of the storage area of the RAM 125 is used as, for example, a work area of the MPU 123.

[0019]The HDD 20 includes a disk 21, a head 22, a spindle motor (SPM) 23, an actuator 24, a driver integrated circuit (IC) 25, a head IC 26, and the main controller 27.

[0020]For example, the disk 21 has a recording surface, in which data is magnetically recorded, on one surface thereof. The disk 21 is rotated fast by the SPM 23. The SPM 23 is driven by a drive current (or drive voltage) which is supplied from the driver IC 25.

[0021]FIG. 1 illustrates a configuration of the HDD 20 including only one disk 21. However, the HDD 20 may have multiple disks 21 which are stacked. In addition, in the configuration of FIG. 1, the disk 21 has a recording surface on one surface only. However, the disk 21 may have recording surfaces on both surfaces and heads may be disposed so as to respectively correspond to both surfaces.

[0022]The disk 21 (in more detail, recording surface of the disk 21) has, for example, multiple concentric tracks. The disk 21 may have multiple tracks which are disposed in a spiral shape. The disk 21 has in advance the system area (SA) 101 in a portion of the recording surface.

[0023]There is a case in which the system area 101 is represented by HDD SA101. Here, management information (HDD management information) on the HDD 20, and information which is the same as management information (NAND management information) on the NAND memory 11 that will be described later are retained (recorded) in the system area 101.

[0024]The head (head slider) 22 is disposed so as to correspond to the recording surface of the disk 21. The head 22 is attached to the tip of a suspension extending from an arm of the actuator 24.

[0025]The actuator 24 includes a voice coil motor (VCM) 240 which becomes a drive source of the actuator 24. The VCM 240 is driven by a drive current (or drive voltage) which is supplied from the driver IC 25. As the actuator 24 is driven by the VCM 240, the head 22 moves in a radial direction of the disk 21 on the disk 21 so as to draw a circular arc.

[0026]The driver IC 25 drives the SPM 23 and the VCM 240 under a control of the main controller 27 (in more detail, the MPU 123 in the main controller 27). As the VCM 240 is driven by the driver IC 25, the head 22 is positioned to a target track on the disk 21.

[0027]The head IC 26 is also called a head amplifier. For example, the head IC 26 is fixed to a predetermined place of the actuator 24, and is electrically connected to the main controller 27 through a flexible printed circuit board (FPC). However, in FIG. 1, for the sake of convenience of drawing, the head IC 26 is disposed in a place separated from the actuator 24.

[0028]The head IC 26 amplifies a signal (that is, a read signal) which is read by a read element of the head 22. In addition, the head IC 26 converts write data which is output from the main controller 27 (in more detail, R/W channel 271 in the main controller 27) into a write current, and outputs the write current to a write element of the head 22.

[0029]The R/W channel 271 performs processing of a signal in relation to read and write. That is, the R/W channel 271 converts a read signal which is amplified by the head IC 26 into digital data, and decodes the read data from the digital data.

[0030]The R/W channel 271 also encodes writ data which is transmitted from the HDC 272, and transmits the encoded write data to the head IC 26.

[0031]The HDC 272 is connected to the host through a host interface (storage interface) 30. The host and the hybrid driver illustrated in FIG. 1 are included in an electronic apparatus such as, a personal computer, a video camera, a music player, a mobile terminal, a mobile phone, or a printer device.

[0032]The HDC 272 receives a signal which is transmitted from the host, and functions as a host interface controller which transmits a signal to the host. In detail, the HDC 272 receives a command (write command, read command or the like) which is transmitted from the host, and transfers the received command to the MPU 123.

[0033]In addition, the HDC 272 controls data transmission between the host and the HDC 272. The HDC 272 further functions as a disk interface controller which controls data writing to the disk 21 and data reading from the disk 21 through the R/W channel 271, the head IC 26, and the head 22.

[0034]The MPU 123 controls access to the NAND memory 11 in accordance with an access requirement (write requirement or read requirement) from the host, and access to the disk 21 through the R/W channel 271, the head IC 26, and the head 22. This kind of control is performed by a second control program. In the present embodiment, the second control program is stored in, for example, the ROM 124. A part of the storage area of the RAM 125 is used as a work area of, for example, the MPU 123.

[0035]An initial program loader (IPL) may be stored in the ROM 124, and the second control program may be stored in the disk 21. In this case, it is preferable that, when a power supply is connected to the hybrid driver, the MPU 123 operates IPL, and thereby the second control program is loaded in the ROM 124 or the RAM 125 from the disk 21.

[0036]FIG. 2 is a conceptual diagram illustrating an example of a format of a storage area of the NAND memory 11 illustrated in FIG. 1. In FIG. 2, the storage area of the NAND memory 11 is configured N (=K+L) programs (that is, physical block). In the NAND memory 11, data is collectively erased by using the data as a unit. That is, block is a unit by which data is erased.

[0037]The storage area of the NAND memory 11 is divided into, for example, the system area (SA) 111 and a cache area (CA) 112, as illustrated in FIG. 1 and FIG. 2. That is, the NAND memory 11 includes the system area 111 and the cache area 112.

[0038]The system area 111 is small enough in general with respect to the cache area 112. The system area 111 of the NAND memory 11 may be referred to as NAND SA111, and the cache area 112 of the NAND memory 11 may be referred to as NAND CA112.

[0039]In addition, in the present embodiment, the system area 111 includes L blocks, and the cache area 112 includes K blocks. Furthermore, as described above, the system area 111 is smaller in general than the cache area 112, and thus, it is assumed that K>L.

[0040]The system area 111 is used to store information (NAND management information) which is used for a system (for example, the main controller 27) to manage processing of data reading, data writing, or data erasing with respect to the NAND memory 11. That is, the NAND management information of the NAND memory 11 is retained in the system area 111.

[0041]It is preferable that the NAND management information is retained redundantly (multiplexed), and thus backup data of the NAND management information may be retained in the system area 111. For example, the cache area 112 is used to store data with high access frequency from the host. Meanwhile, the cache area 112 may store data with a high access possibility from the host, and may store data which is recently accessed by the data.

[0042]The NAND management information includes information of physical configuration of the NAND memory 11, the number of commands (for example, erasing) which is processed with respect to the NAND memory 11, the number of data which is rewritten to the NAND memory 11 as described above, or the like.

[0043]since the minimum unit of writing and the minimum unit of erasing are different from each other in the storage area of the NAND memory 11, only a portion of data cannot be rewritten. For example, in the NAND memory 11, the minimum unit of writing is one page, and the minimum unit of erasing is one block. For example, one block includes 64 pages, but is not limited to this.

[0044]An erasing operation of the storage area of the NAND memory 11 is performed by a block unit including multiple pages as described above. In addition, rewriting (overwriting) operation is not completed by one operation, and data writing is performed after erasing. That is, since it is necessary to erase the entirety of one block even when one page is rewritten, data of the one block is temporarily retained in another storage area.

[0045]The NAND management information retained in the system area 111 is acquired when the hybrid drive starts up (power supply is connected). If the NAND management information cannot be acquired, the entire data in the NAND memory 11 is treated as lost data. One of causes that the NAND management information cannot be acquired is degradation of the storage area (particularly, system area 111) of the NAND memory 11.

[0046]A plurality of non-volatile storage media are mounted in the hybrid driver. For example, as a plurality of NAND memories are provided in the hybrid driver, the degradation of the system area of the NAND memory can be suppressed to some extent. However, in this case, as long as an enough number of NAND memories are not mounted, it is not possible to multiplex an enough number of system areas. As a result, degradation of the system area is not sufficiently suppressed.

[0047]There is a possibility that the NAND management information cannot be acquired by degradation of the system area in the hybrid driver. If the NAND management information cannot be acquired, reliability of the operation of the hybrid driver is decreased.

[0048]For example, an appropriate margin area according to a necessary amount for the NAND management information may be provided in the system area 111, as degradation countermeasure for the storage area of the NAND memory 11. The NAND management information can also be stored in the margin area in the system area 111. As a result, usage concentration of a specified area of the system area 111 can be avoided, writing to the system area 111 can be smoothed, and degradation of the storage area can be reduced.

[0049]The system area 111 is used to store, for example, a logical-physical conversion table, a first free area list, a second free area list, and a bad block list. In the following description, the logical-physical conversion table may be referred to simply as a table. In addition, the first free area list, the second free area list, and the bad block list may be referred to simply as a list.

[0050]The logical-physical conversion table is used to store block management information for managing each block in the cache area 112 of the NAND memory 11. In the embodiment, the block management information is used as cache directory information on addresses of data (each block data) stored in each block (area with a predetermined size) in the cache area 112.

[0051]The cache directory information includes information for managing correspondence between a physical address and logical address of each block data. The physical address (here, physical block number) of each block data indicates a position of a block (area) in the NAND memory 11 in which each block data is stored. The logical address (here, a logical block number) of each block data indicates a position in a logical address space of each block data. In general, in the NAND memory, if both the NAND management information and the logical-physical conversion table are not read, preparation of the start-up is not completed as will be illustrated later.

[0052]The first free area list is used to register a free area of a first type in the cache area 112. That is, the first free area list is used as first information for managing the free area of a first type. The free area of a first type indicates a normal free area.

[0053]The second free area list is used to register a free area of a second type in the cache area 112. That is, the second free area list is used as second information for managing the free area of a second type. The free area of a second type indicates a free area in which a read error has occurred before. The bad block list is used to register an unusable block (physical block), that is, a bad block (area). That is, the bad block list is used as third information for managing bad blocks.

[0054]As described above, in the NAND memory 11, new data (updated data) cannot be overwritten to an area in which data is stored in advance. For this reason, the stored position (memory position) of the table in the system area 111 can be changed whenever the table is updated. In this case, the updated table (new table) can be written to an area different from the area in which a table (old table) before being updated. The stored position of the list in the system area 111 is the same as above.

[0055]Information on the stored position and a size of the table, the list, or the like in the system area 111 is stored in a part of, for example HDD SA101, NAND SA111, or the like. In the present embodiment, the information stored in HDD SA101, NAND SA111, or the like is read when a power supply is connected to the hybrid driver 1, and is loaded in the RAM 125.

[0056]It is assumed that each block of the cache area 112 is configured by multiple pages (physical page). In this case, the logical block is also configured by multiple pages (logical pages).

[0057]A logical page number indicates a logical page (logical page in the logical block) to which pages (physical pages) of a corresponding physical block number and a corresponding physical page number are assigned. That is, the logical page number indicates a position of a logical address space of data which is stored in the corresponding physical page.

[0058]In the present embodiment, the main controller 27 performs read patrol with respect to the NAND memory 11 (in more detail, with respect to the data retained in each block of the NAND memory 11). The read patrol (first processing) according to the present embodiment will be hereinafter described.

[0059]In general, the NAND memory 11 has an upper limit with respect to data rewriting. In addition, retaining period of storage content is also limited, and the storage content can be lost by degradation of the NAND memory 11, if a predetermined period passes. Meanwhile, the predetermined period is, for example, 10 years, but is not limited to this. In addition, the retaining period of the storage content of the NAND memory 11 is shortened by repeated data writing as described above. In addition, it is also known that the storage content of the retaining period is shortened if the NAND memory 11 is used under high temperature environment.

[0060]Therefore, the main controller 27 confirms whether or not data is correctly read, by periodically reading (read patrol is performed) the data retained in each block Block(n) (However, 0£n£N-1) of the NAND memory 12. Meanwhile, in the present embodiment, the read patrol is periodically performed by, for example, a predetermined periodic cycle.

[0061]In the present embodiment, “read patrol” is an operation of reading data retained in each block Block(n), and confirming whether or not the read data is correctly read, as described above. In other words, the read patrol is an operation of confirming whether or not the data retained in each block Block(n) is damaged.

[0062]FIG. 3 is a flowchart illustrating an operation of read patrol with respect to the NAND memory 11 according to the embodiment. Hereinafter, an operation of the storage device 1 will be described with reference to FIG. 3. In the read patrol, the main controller 27 performs data reading from each block of the NAND memory 11, as described above, but, here, it is assumed that the data reading is performed from the block Block(n=0).

[0063]The main controller 27 reads first data retained in the block Block(0) of the NAND memory 11 (S101). Meanwhile, an error correction code (ECC) is attached in general to the data retained in the NAND memory 11.

[0064]The ECC is a code (error correction code) for correcting an error, in a case in which the error occurs in the data when data is read. Meanwhile, a rate of an error in each data is called an error rate. For example, the error rate indicates a rate of error bits with respect to the number of entire bits of data.

[0065]In addition, error correction of ECC has an upper limit, and in a case in which the number of error bits is quite great, that is, if the error rate is quite great, the ECC may not correct the error. That is, there is an upper limit of the number (correctable rate) of correctable bits in the ECC. Meanwhile, in a case in which the ECC has a high upper limit of the number (correctable rate) of correctable bits, it may be expressed that correction strength of the ECC is high (strong).

[0066]Subsequently, an error rate of data which is read in step S101 is acquired (S102). Meanwhile, here, the main controller 27 acquires an error rate of the data which is read from the block Block(0), but may acquire, for example, the number of error bits.

[0067]That is, in the present embodiment, the main controller 27 acquires an error amount of the data which is read. Meanwhile, the error amount includes an error rate and the number of error bits. Meanwhile, in the following description, it is assumed that the main controller 27 acquires an error rate of the data which is read.

[0068]The main controller 27 confirms whether or not the error rate acquired in step S102 is greater than a first predetermined value th1 (threshold, a first value) (S103).

[0069]In step S103, if error rate>th1 (S103:Yes), the main controller 27 confirm whether or not the error rate acquired in step S102 is greater than a second predetermined value th2 (threshold, a second value) (S104). However, th1<th2 is satisfied.

[0070]Meanwhile, if error rate£th1 (S103:No), processing proceeds to step S110. An operation after step S110 will be described later.

[0071]In step S104, if error rate>th2 (S104:Yes), the main controller 27 confirms whether or not data in the block Block(0) can be correctly read (S107). In other words, the main controller 27 confirms whether or not the error of the data in the block Block(0) is (can be) corrected by the ECC.

[0072]In step S107, if data reading is correctly performed (S107:Yes), the main controller 27 writes the data which is read from the block Block(0) to a free block of the NAND memory 11 (S109), and updates the logical-physical conversion table. Meanwhile, a free block is a block in which valid data is not retained. In addition, the free block is the free area of a first type described above. Thereafter, processing proceeds to step S110.

[0073]Meanwhile, if the data reading is not correctly performed in step S107 (S107:No), the main controller 27 writes data corresponding to the data which is not correctly read to a free block (free area of a first type) of the NAND memory 11 (S108), in a case in which the data corresponding to the data which is not correctly read exists in the data retained in the disk 21, and updates the table. Thereafter, processing proceeds to step S110.

[0074]In addition, if an answer of the processing is No (S107:No), the data retained in the block Block(0) is erased or becomes invalid, and the block Block(0) becomes a free block. Meanwhile, the free block in this case is the free area of a second type.

[0075]Subsequently, a case in which error rate£th2 (S104:No) in step S104 will be described. If an answer of the processing in step S104 is No (S104:No), the main controller 27 confirms whether or not the data in the block Block(0) of the NAND memory 11 is dirty data (S105). The dirty data indicates data which is written to the NAND memory 11 (in more detail, the cache area 112 of the NAND memory 11), and is not written to the disk 21.

[0076]In step S105, if the data in the block Block(0) is dirty data (S105:Yes), the main controller 27 writes the data to the disk 21 (S106). Thereafter, the processing proceeds to step S110.

[0077]Meanwhile, in step S105, if the data in the block Block(0) is not dirty data (S105:No), the processing proceeds to step S110.

[0078]If answers of the processing in steps S103 and S105 are no (S103:No) and (S105:No), and after the processing of steps S106, S108, and S109 is completed, a variable n becomes (n+1) (n=n+1) (S110). That is, until now, n is zero (n=0) but n will be one (n=1), and a target block is changed from the block Block(0) to the block Block(1).

[0079]Thereafter, the main controller 27 confirms whether or not the value of n is greater than the number of blocks (N=K+L) of the NAND memory 11 (S111).

[0080]If n>N (S111:Yes), data recognition is completed with respect to the entire blocks of the NAND memory 11, and thus the main controller 27 completes the read patrol.

[0081]Meanwhile, if n£N (S111:No), the processing returns to step S101, and data retained in the next block is read. That is, the main controller 27 reads data in the block Block(n=1), and thereafter, the processing which is the same as that described above is performed.

[0082]In the present embodiment, the main controller 27 performs data reading and confirming (read and verify) with respect to each block of the NAND memory 11. In addition, at the time of the read patrol, a data saving destination is selected by using the first predetermined value th1 having a sufficient margin with respect to the correctable rate of ECC, and the second predetermined value th2 which is greater than the first predetermined value th1, as thresholds.

[0083]Specifically, if an error rate of the data read from a certain block Block(n) of the NAND memory 11 is greater than the second predetermined value th2, the main controller 27 writes the data to another free block of the NAND memory 11. If an error rate of the data read from the block Block(n) is greater than the first predetermined value th1 and is less than or equal to the second predetermined value th2, the main controller 27 writes the data to the disk 21.

[0084]As described above, the NAND memory 11 has an upper limit for the number of data rewritings, and is degraded in accordance with an increase of the number of data rewritings. Thus, it is preferable to suppress an increase of the number of data rewritings.

[0085]In the present embodiment, an error rate of the data which is read is greater than the first predetermined value th1 having a sufficient margin with respect to a correctable rate of ECC. However, if the error rate of the data is less than or equal to the second predetermined value th2>th1, the data which is read is written to the disk 21, if the data is dirty data.

[0086]For this reason, an increase of the number of rewritings to the NAND memory 11 can be suppressed, and degradation of the NAND memory 11 can be decreased, compared to a case in which data is written to another free area of the NAND memory 11.

[0087]In addition, if an error rate of data which is correctly read is greater than the first predetermined value th1 and the second predetermined value th2 with respect to a correctable rate of ECC, the data which is read is written to another free block of the NAND memory 11.

[0088]For this reason, it is possible to prevent performance from degrading, even if there is a high possibility that a read error occurs, compared to a case in which data is written to the aforementioned disk 21. That is, since data is saved in the NAND memory 11 when a read error occurs due to data reading from the NAND memory 11, it is not necessary to access the disk 21, and it is possible to maintain fast data reading.

[0089]In addition, as another example of degradation countermeasure of the NAND memory 11, a plurality of the NAND memories 11 are provided, each of the NAND memories 11 has the system area 111, and thus it is possible to multiplex data to be recorded. However, in this case, it is necessary to provide a sufficient number of the NAND memories 11, and additional cost can be required.

[0090]Meanwhile, in the present embodiment, it is possible to select the data writing destination from another free block of the NAND memory 11, and the disk 21, in accordance with an error rate of data which is read, and thus, degradation of the NAND memory 11 and performance degradation of data access are suppressed. For this reason, it is not necessary to increase the number of the NAND memories 11, or to increase capacity. Accordingly, it is possible to reduce cost and an entire size of the storage device 1.

[0091]Meanwhile, the read patrol which is performed in the present embodiment is periodically performed in a predetermined cycle, while power is supplied to, for example, the NAND memory 11, but the read patrol may be configured so as to be appropriately performed in accordance with a command from the host. In this case, it is possible to control timing for the read patrol in accordance with usage state of the NAND memory 11 or a load of the main controller 27.

[0092]Furthermore, it is possible to provide a configuration in which power supply to the NAND memory 11 is diskonnected after termination of power supply to the NAND memory 11 is notified to the main controller 27 before the power supply is terminated, and the read patrol is performed in accordance with input from the host.

[0093]By providing the configuration, when the NAND memory 11 is used next time, it is possible to use the NAND memory 11 in a state in which the data retained in the NAND memory 11 is confirmed.

[0094]In addition, the main controller 27 may be configured such that the read patrol is initially performed when power supply to the NAND memory 11 is started. Meanwhile, in this case, “initially performing the read patrol” means that the main controller 27 performs the read patrol before processing according to a command is started after receiving the command such as a write requirement or a read requirement from the host. Thus, it is not necessary to perform the read patrol shortly after power is supplied to the NAND memory 11.

[0095]Even in this configuration, when the NAND memory 11 is used, it is possible to use the NAND memory 11 in a state in which data retained in the NAND memory 11 is confirmed in advance.

[0096]Meanwhile, in the present embodiment, the data writing destination at the time of read patrol is selected in a state in which two values of the first predetermined value th1 and the second predetermined value th2 are set as thresholds, but the invention is not limited to this, and the threshold may be set to, for example, three or more.

[0097]In addition, in the present embodiment, selection of data writing destination in which two values of the first predetermined value th1 and the second predetermined value th2 at the time of read patrol are set as thresholds, is described. However, a series of processing illustrated in FIG. 3 may be performed at the time of reading according to a read requirement from, for example, the host, and may not be performed only at the time of read patrol.

[0098]Furthermore, the selection of the data writing destination described in the present embodiment does not need to be performed all the time. For example, exhaustion of the NAND memory 11 may exceed a determined value, and environment temperature around the NAND memory 11 (or storage device 1) may exceed a determined value.

[0099]In addition, in the present embodiment, the main controller 27 controls the NAND memory 11 and the disk 21, but the invention is not limited to this, and the NAND memory 11 and the disk 21 may be respectively controlled by controller different from each other. In this case, the main controller (control unit) 27 according to the present embodiment includes the respective controllers described above.

[0100]While certain embodiments have been described, these embodiments have been presented by way of example only, and are not intended to limit the scope of the inventions. Indeed, the novel embodiments described herein may be embodied in a variety of other forms; furthermore, various omissions, substitutions and changes in the form of the embodiments described herein may be made without departing from the spirit of the inventions. The accompanying claims and their equivalents are intended to cover such forms or modifications as would fall within the scope and spirit of the inventions.

WHAT IS CLAIMED IS:

1. A storage device comprising:

a first non-volatile storage medium;

a second non-volatile storage medium which has a plurality of blocks and can perform data processing at a faster speed than the first non-volatile storage medium; and

a control unit which includes a first controller for controlling the first non-volatile storage medium, and a second controller for controlling the second non-volatile storage medium,

wherein the control unit acquires an error amount of first data which is read from the second non-volatile storage medium, and writes the read first data onto one of the first non-volatile storage medium and the second non-volatile storage medium in accordance with the error amount.

2. The device according to Claim 1, wherein the control unit writes the first data which is read onto the first non-volatile storage medium, if the error amount is greater than a first value and is less than or equal to a second value greater than the first value, and reads the first data which is read to the second non-volatile storage medium, if the error amount is greater than the second value.

3. The device according to Claim 2, wherein the control unit writes the first data which is read from a first block of the second non-volatile storage medium onto a second block, if the error amount is greater than the second value.

4. The device according to Claim 3, wherein the second block is a free block.

5. The device according to any one of Claims 1 to 4,

wherein the control unit performs first processing with respect to each of the plurality of blocks of the second non-volatile storage medium, and

wherein in the first processing, data which is retained in each of the plurality of blocks is read, and an error amount of the data is acquired.

6. The device according to Claim 5, wherein the first processing is periodically performed in a predetermined cycle.

7. The device according to Claim 5, wherein the first processing is performed in accordance with an instruction from the outside.

8. The device according to Claim 5, wherein the first processing is performed earlier than the other processing in accordance with a command from a host, if power supply to the second non-volatile storage medium is started.

9. The device according to Claim 5, wherein the first processing is performed, if the control unit receives an input from a host indicating that power supply to a second storage medium is disconnected.

10. A method of a storage device which includes a first non-volatile storage medium and a second non-volatile storage medium that can perform data processing faster than the first non-volatile storage medium, the method comprising:

acquiring an error amount of first data which is read from the second non-volatile storage medium, and

writing the first data which is read onto one of the first non-volatile storage medium and the second non-volatile storage medium in accordance with the error amount.

11. The method according to Claim 10,

wherein, if the error amount is greater than a first value and is less than or equal to a second value greater than the first value, the first data which is read is written onto the first non-volatile storage medium, and

wherein, if the error amount is greater than the second value, the first data which is read is written onto the second non-volatile storage medium.

ABSTRACT

According to one embodiment, a storage device includes a first non-volatile storage medium, a second non-volatile storage medium which has a plurality of blocks and can perform data processing at a faster speed than the first non-volatile storage medium, and a control unit which includes a first controller for controlling the first non-volatile storage medium, and a second controller for controlling the second non-volatile storage medium, in which the control unit acquires an error amount of first data which is read from the second non-volatile storage medium, and writes the read first data onto one of the first non-volatile storage medium and the second non-volatile storage medium in accordance with the error amount.

Drawings

FIG. 1

27: MAIN CONTROLLER

122: MEMORY IF

11: NAND MEMORY

HOST

271: R/W CHANNEL

26: HEAD IC

25: DRIVER IC

FIG. 3

S102: ACQUIRE ERROR RATE

S105: IS NAND BLOCK Block(n) DIRTY DATA?

S106: WRITE TO DISK

S107: CAN NAND BLOCK Block(n) BE READ?

S108: WRITE DISK DATA TO FREE BLOCK OF NAND

S109: WRITE TO FREE BLOCK OF NAND

S111: n>NUMBER OF NAND BLOCK?