

NETWORK STORAGE DEVICE STORING LARGE AMOUNT OF DATA

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims priority to Korean Patent Application No. 10-
5 2018-0044420 filed on April 17, 2018, in the Korean Intellectual Property Office, the
disclosures of which are incorporated by reference herein in their entireties.

TECHNICAL FIELD

[0002] Embodiments of the inventive concept disclosed herein relate to a
10 semiconductor memory device, and more particularly, relate to a network storage
device storing a large amount of data.

BACKGROUND

[0003] An example of a flash memory based mass storage device is a solid
15 state drive (hereinafter referred to as an “SSD”). The use of the SSD diversifies as the
demand on the SSD explosively increases. For example, the SSD is subdivided into an
SSD for server, an SSD for client, an SSD for data center, etc. An SSD interface should
provide the best speed and reliability to be suitable for the subdivided use. For the
purpose of satisfying the requirement, a nonvolatile memory express (NVMe) which
20 is based on serial advanced technology attachment (SATA), serial attached SCSI
(SAS), or peripheral component interconnect express (PCIe) is actively developed and
applied.

[0004] Nowadays, SSD interfaces for making expandability easy in a system
such as a large-capacity memory center are actively being developed. In particular, an
25 NVMe over fabrics (NVM-oF) is actively being developed as a standard for mounting

an SSD on a network fabric such as an Ethernet switch. The NVM-oF supports an NVMe storage protocol through extensive storage networking fabrics (e.g., an Ethernet, a fiber channel, and InfiniBand).

[0005] The NVMe storage protocol is also applied to the NVMe SSD.

5 Accordingly, an interface of storage including the NVMe SSD translates a protocol for a network fabric into the NVMe-oF protocol or has a function of a data buffer. However, in this case, since there is a need for protocol translation corresponding to a plurality of protocol layers, an increase in latency is inevitable.

[0006] Accordingly, to reduce a latency and make capacity expansion easy,
10 there is a need for a technology for using not the NVMe SSD but a nonvolatile memory device or a nonvolatile memory array as a storage medium and performing multi-protocol translation efficiently.

SUMMARY

15 **[0007]** Embodiments of the inventive concept provide an Ethernet storage device for simplifying a controller structure of a network storage device and providing the ease of expansion.

[0008] According to an exemplary embodiment, a network storage device connected with a network fabric includes a network storage controller that performs
20 interfacing with the network fabric and translates and executes a command provided through the network fabric, and a nonvolatile memory cluster that exchanges data with the network storage controller under control of the network storage controller. The nonvolatile memory cluster includes a first nonvolatile memory array connected with the network storage controller through a first channel, a nonvolatile memory switch
25 connected with the network storage controller through a second channel, and a second

nonvolatile memory array communicating with the network storage controller under control of the nonvolatile memory switch.

[0009] According to an exemplary embodiment, a network storage device includes a storage controller that translates a first command provided through a network switch into a second command for controlling a nonvolatile memory and outputs the second command by using an interface, and a nonvolatile memory cluster that stores write data depending on the second command output from the storage controller or outputs stored data to the storage controller. The storage controller includes one or more external computing devices that perform a command for the nonvolatile memory cluster.

BRIEF DESCRIPTION OF THE FIGURES

[0010] The above and other objects and features of the inventive concept will become apparent by describing in detail exemplary embodiments thereof with reference to the accompanying drawings.

[0011] FIG. 1 is a block diagram illustrating a network storage system according to an embodiment of the inventive concept.

[0012] FIG. 2 is a block diagram illustrating an embodiment of a network storage controller of FIG. 1.

[0013] FIG. 3 is a block diagram illustrating a configuration of a storage controller of FIG. 2.

[0014] FIG. 4 is a block diagram illustrating another embodiment of a network storage controller of FIG. 1.

[0015] FIG. 5 is a block diagram illustrating a network storage controller according to another embodiment of the inventive concept.

[0016] FIG. 6 is a block diagram illustrating a network storage controller according to another embodiment of the inventive concept.

[0017] FIG. 7 is a diagram illustrating a part of an NVM cluster of the inventive concept.

5 **[0018]** FIG. 8 is a diagram illustrating another part of an NVM cluster of the inventive concept.

[0019] FIG. 9 is a diagram illustrating another part of an NVM cluster of the inventive concept.

[0020] FIG. 10 is a block diagram illustrating a network storage system
10 configurable according to an embodiment of the inventive concept.

[0021] FIG. 11 is a block diagram illustrating a network storage system according to an embodiment of the inventive concept.

DETAILED DESCRIPTION

15 **[0022]** It should be understood that both the foregoing general description and the following detailed description are provided as examples, and it should be regarded as an additional description is provided. Reference numerals will be represented in detail in embodiments of the inventive concept, examples of which are illustrated in the accompanying drawings. Wherever possible, the same reference numerals are used
20 in the drawings and the description to refer to the same or similar parts.

[0023] Below, a solid state drive (SSD) using a flash memory device will be used as an example of a storage device for describing the features and functions of the inventive concept. However, one skilled in the art may easily understand other merits and performance of the inventive concept depending on the contents disclosed here.
25 The inventive concept may be implemented or applied through other embodiments. In

addition, the detailed description may be changed or modified according to view points and applications without departing from the claims, the scope and spirit, and any other purposes of the inventive concept.

[0024] FIG. 1 is a block diagram illustrating a network storage system according to an embodiment of the inventive concept. Referring to FIG. 1, a network storage system includes an Ethernet switch 100 and a network storage device 200. A host (not illustrated) transmits a command and data of an Ethernet protocol to the network storage device 200 through the Ethernet switch 100. The network storage device 200 may translate the command and data of the Ethernet protocol into a command and data to be transmitted directly to an NVM cluster 280 without translation into an intermediate layer (e.g., an NVMe protocol).

[0025] The Ethernet switch 100 provides a command or data provided from the host to the network storage device 200. Alternatively, the Ethernet switch 100 transmits data, which the network storage device 200 provides depending on a request of the host, to the host. That is, the Ethernet switch 100 may be a network fabric or switch which uses an Ethernet protocol. When transmitting a command and data to the network storage device 200, the Ethernet switch 100 may transmit the command and the data in compliance with the Ethernet protocol including an NVMe-oF storage protocol. Also, the Ethernet switch 100 may exchange a response or data with the network storage device 200 in compliance with the Ethernet protocol.

[0026] The network storage device 200 controls the NVM cluster 280 in response to the command CMD or the data provided through the Ethernet switch 100. The NVM cluster 280 may include NVM switches 230, 232, and 234, and nonvolatile memory devices 240, 250, 260, and 270. The network storage device 200 may directly translate a command or data of an Ethernet protocol format provided from the Ethernet

switch 100 to a command or data format for controlling the nonvolatile memory devices 240, 250, 260, and 270. For the purpose of performing the function, the network storage device 200 includes a network storage controller 210. In the network storage controller 210, transmission formats for supporting the Ethernet protocol, the NVMe-oF protocol, and the NVMe protocol may be processed at a single layer. The network storage controller 210 may be implemented with a single chip.

[0027] The network storage controller 210 provides interfacing between the Ethernet switch 100 and the network storage device 200. The network storage controller 210 may directly translate a command or data of a network transmission format provided from the Ethernet switch 100, for example, an Ethernet protocol format (e.g., a packet) to a command or data format to be applied to the nonvolatile memory devices 240, 250, 260, and 270. In the network storage controller 210, transmission formats for supporting the Ethernet protocol, the NVMe-oF protocol, and the NVMe protocol may be processed at a single layer. An operation of the network storage controller 210 will be more fully described later.

[0028] A buffer memory 220 may be used as a data buffer for data exchange between the network storage device 200 and the Ethernet switch 100. For example, the buffer memory 220 may be implemented with a dynamic random access memory (DRAM). However, the buffer memory 220 is not limited to the DRAM. For example, the buffer memory 220 may be implemented with a volatile memory, such as a static RAM (SRAM) or a synchronous DRAM (SDRAM), a nonvolatile RAM NV-RAM, or a combination thereof.

[0029] The NVM cluster 280 may include the NVM switches 230, 232, and 234, and the nonvolatile memory devices 240, 250, 260, and 270. Here, each of the nonvolatile memory devices 240, 250, 260, and 270 may include a nonvolatile memory

array (NVM array). That is, the nonvolatile memory device 240 may include one or more NAND flash memory devices, and the remaining nonvolatile memory devices 250, 260, and 270 may also be implemented or combined in a similar form.

[0030] The NVM switches 230, 232, and 234 are provided as a configuration for expansion-connecting a nonvolatile memory device to a flash interface of the network storage controller 210. For example, each of the NVM switches 230, 232, and 234 may independently perform address mapping on a nonvolatile memory device connected thereto. A structure of the NVM switches 230, 232, and 234 and nonvolatile memory devices connected with the NVM switches 230, 232, and 234 will be more fully described with reference to the following drawings.

[0031] According to the above description, the network storage device 200 of the inventive concept includes the network storage controller 210 which may directly translate a network protocol into a command or data format for controlling a nonvolatile memory device. Accordingly, the network storage controller 210 may load/store a command and data transmitted from the network fabric to the nonvolatile memory devices 240, 250, 260, and 270 after being processed through a command path and a data path, which are separated from each other. Also, the nonvolatile memory devices 240, 250, 260, and 270 may be easily expanded by using an NVM switch.

[0032] FIG. 2 is a block diagram illustrating an embodiment of a network storage controller of FIG. 1. Referring to FIG. 2, a network storage controller 210a according to an embodiment may include a storage controller 214.

[0033] The storage controller 214 performs interfacing between the Ethernet switch 100 and the NVM cluster 280 by using an embedded CPU 211. In particular, the storage controller 214 is connected with the Ethernet switch 100 through a first

port PT1. For example, the first port PT1 may perform a network interface function for a connection with the Ethernet switch 100.

[0034] The storage controller 214 may exchange data with the buffer memory 220 through a second port PT2. The storage controller 214 is connected with the NVM cluster 280 through a third port PT3. Here, the third port PT3 may include a plurality of channels CH1, CH2, and CH3 as illustrated in FIG. 1. Through at least one of the channels CH1, CH2, and CH3 which the third port PT3 provides, the storage controller 214 may be directly connected with a nonvolatile memory (NVM) array of the NVM cluster 280. Alternatively, through at least one of the channels CH1, CH2, and CH3 which the third port PT3 provides, the storage controller 214 may be connected with the NVM switches 230 and 232. Alternatively, through at least one of the channels CH1, CH2, and CH3 which the third port PT3 provides, the storage controller 214 may be connected with cascade NVM switches.

[0035] FIG. 3 is a block diagram illustrating an exemplary configuration of a storage controller of FIG. 2. Referring to FIG. 3, the storage controller 214 includes the embedded CPU 211, a working memory 213, a host interface 215, a buffer manager 217, and a flash interface 219.

[0036] The embedded CPU 211 may transmit a variety of control information, which is needed to perform a read/write operation associated with the NVM cluster 280, to registers of the host interface 215 and the flash interface 219. The embedded CPU 211 may operate based on firmware or an operating system OS provided for various control operations of the network storage controller 210. For example, the embedded CPU 211 may execute garbage collection for managing the NVM cluster 280, or a flash translation layer (FTL) for performing address managing, wear leveling,

etc. In particular, the embedded CPU 211 may call and execute a storage manager 216 loaded to the working memory 213.

[0037] As the storage manager 216 is executed, the embedded CPU 211 may process transmission formats for supporting the Ethernet protocol, the NVMe-oF protocol, and the NVMe protocol with respect to a command or data provided through the Ethernet switch 100, at a single layer. In addition, the embedded CPU 211 may load/store a command and data transmitted from a network fabric to the NVM cluster 280 after being processed through a command path and a data path, which are separated from each other.

10 **[0038]** The working memory 213 may be used as an operation memory, a cache memory, or a buffer memory of the embedded CPU 211. The working memory 213 may store codes and commands which the embedded CPU 211 executes. The working memory 213 may store data processed by the embedded CPU 211. In an embodiment, the working memory 213 may be implemented with a static random access memory (SRAM). In particular, the storage manager 216 may be loaded to the working memory 213. When executed by the embedded CPU 211, the storage manager 216 may process translation of a transmission format of a command or data provided from the Ethernet switch 100 at a single layer. In addition, the storage manager 216 may process a command or data transmitted from the network fabric in a state where
15 a command path and a data path are separated. In addition, the flash translation layer FTL or various memory management modules may be stored in the working memory 213.

[0039] The storage manager 216 may collect and adjust pieces of overall information about the NVM cluster 280. That is, the storage manager 216 may
25 maintain and update a status or mapping information of data stored in the NVM cluster

280. Accordingly, even though an access request is made from the Ethernet switch 100, the storage manager 216 may provide requested data at high speed to the Ethernet switch 100 or may write write-requested data at high speed. In addition, since the storage manager 216 has the authority to manage a mapping table for managing an address of data, the storage manager 216 may perform data migration within the NVM cluster 280 or correction of mapping information if necessary.

[0040] The host interface 215 may communicate with the Ethernet switch 100. The host interface 215 may recognize a command of a network transmission format provided from the Ethernet switch 100. For example, the host interface 215 may communicate with the Ethernet switch 100 by using a communication protocol of a high-speed Ethernet system, such as a Fibre channel or InfiniBand. The host interface 215 may translate a command of a network transmission format transmitted from the Ethernet switch 100 and may provide the translated command to the flash interface 219. Alternatively, the host interface 215 may translate data provided through the flash interface 219 so as to include the network transmission format, and may transmit the translated data to the Ethernet switch 100.

[0041] The buffer manager 217 may control read and write operations of the buffer memory 220 (refer to FIG. 1). For example, the buffer manager 217 temporarily stores write data or read data in the buffer memory 220. The buffer manager 217 may classify and manage a memory area of the buffer memory 220 in units of streams under control of the embedded CPU 211.

[0042] The flash interface 219 exchanges data with the NVM cluster 280. The flash interface 219 writes data transmitted from the buffer memory 220 to the NVM cluster 280 through the respective memory channels CH1, CH2, and CH3. Read data which are read out from the NVM cluster 280 and are provided through a memory

channel may be collected by the flash interface 219. Afterwards, the collected data may be stored to the buffer memory 220.

[0043] The storage controller 214 of the above-described structure may translate a network protocol for communication with the Ethernet switch 100 directly to a command or data of a flash memory level. Accordingly, a command or data provided through the network fabric may not experience a plurality of sequential translation processes, which are performed through, for example, an Ethernet network interface card (NIC), a TCP/IP offload engine, and a PCIe switch. According to the above-described feature, a command or data transmitted from the Ethernet switch 100 may be loaded/stored to the NVM cluster 280 after being processed through a command path and a data path, which are separated from each other.

[0044] In particular, the storage controller 214 may be implemented with a single chip. As the storage controller 214 is implemented with a single chip, the network storage device 200 of the inventive concept may be lightweight, thin, and small-sized. Accordingly, the network storage device 200 of the inventive concept may provide a low latency, high economic feasibility, and high expandability on the network fabric.

[0045] FIG. 4 is a block diagram illustrating another embodiment of a network storage controller of FIG. 1. Referring to FIG. 4, a network storage controller 210b according to another embodiment may include an external processor 211b and a storage controller 214b.

[0046] The storage controller 214b may perform interfacing between the Ethernet switch 100 and the NVM cluster 280 by using the external processor 211b. The external processor 211b may translate a command or data of an Ethernet protocol transmitted from the Ethernet switch 100. In addition, the external processor 211b may

control the storage controller 214b so as to search for data stored in the NVM cluster 280 or to output the found data to the Ethernet switch 100 depending on a request provided through the Ethernet switch 100. The external processor 211b may include one or more processors specialized for network management.

5 **[0047]** The external processor 211b may perform a function of an embedded processor included in the storage controller 214b, and may perform distributed processing on a workload with an embedded processor. In addition, the external processor 211b may perform processing of a separate operation requested from a host, such as network management.

10 **[0048]** The storage controller 214b is electrically connected with the Ethernet switch 100 through the first port PT1. The storage controller 214b may exchange data with the buffer memory 220 through the second port PT2. The storage controller 214b is connected with the NVM cluster 280 through the third port PT3. Here, the third port PT3 may include the plurality of channels CH1, CH2, and CH3 as illustrated in FIG.

15 1. In addition, the storage controller 214b may be connected with the external processor 211b through a fourth port PT4. The storage controller 214b may perform an operation of translating a protocol, an operation of searching for data stored in the NVM cluster 280, or an operation of loading/storing data to the NVM cluster 280, by using the external processor 211b. The fourth port PT4 may be implemented with a

20 PCIe interface which may be connected with a plurality of processors. However, the inventive concept is not limited thereto. The storage controller 214b may have a configuration and a function similar to those of the storage controller 214 of FIG. 3 except that the storage controller 214b is controlled by the external processor 211b.

Here, the storage controller 214b may be implemented with one chip, and the network
25 storage controller 210b may be implemented on one PCB board.

[0049] FIG. 5 is a block diagram illustrating a network storage controller 210c according to another embodiment of the inventive concept. Referring to FIG. 5, a network storage controller 210c may be implemented with one chip.

[0050] The network storage controller 210c includes the first port PT1
5 connected with the Ethernet switch 100. The first port PT1 may be a hardware port for connecting the network storage controller 210c with a high-speed Ethernet system such as a fiber channel or InfiniBand. The first port PT1 may transmit a command or data of the Ethernet protocol to the host interface 215. The host interface 215 may perform communication with the Ethernet switch 100 in compliance with the Ethernet
10 protocol. For example, the communication protocol of the high-speed Ethernet system, such as a fiber channel or InfiniBand, may be translated to a transmission format for controlling the NVM cluster 280 by the host interface 215.

[0051] The buffer manager 217 may control read and write operations of the buffer memory 220 (refer to FIG. 1). For example, the buffer manager 217 temporarily
15 stores write data or read data to the buffer memory 220. The buffer manager 217 may classify and manage a memory area of the buffer memory 220 in units of streams under control of an embedded CPU 211c.

[0052] The flash interface 219 exchanges data with the NVM cluster 280. The flash interface 219 writes data transmitted from the buffer memory 220 to the NVM
20 cluster 280 through respective channels. Read data which are read out from the NVM cluster 280 and are provided through a memory channel may be collected by the flash interface 219. Afterwards, the collected data may be stored in the buffer memory 220.

[0053] The embedded CPU 211c may transmit a variety of control information, which is needed to perform a read/write operation associated with the
25 NVM cluster 280, to registers of the host interface 215 and the flash interface 219. The

embedded CPU 211c may operate based on firmware or an operating system OS provided for various control operations of the network storage controller 210c. For example, the embedded CPU 211c may execute garbage collection for managing the NVM cluster 280, or a flash translation layer (FTL) for performing address managing, wear leveling, etc. In particular, the embedded CPU 211c may call and execute an operating system OS or a management module loaded to the working memory 213. The embedded CPU 211c may process transmission formats for supporting the Ethernet protocol, the NVMe-oF protocol, and the NVMe protocol with respect to a command or data provided to the network storage controller 210c, at a single layer.

10 **[0054]** As described above, the network storage controller 210c according to another embodiment may directly perform network interfacing and may include the embedded CPU 211c. Accordingly, the network storage controller 210c may be directly connected to the Ethernet switch 100 and may control the NVM cluster 280.

15 **[0055]** FIG. 6 is a block diagram illustrating a network storage controller 210d according to another embodiment of the inventive concept. Referring to FIG. 6, the network storage controller 210d may perform interfacing between the NVM cluster 280 and the Ethernet switch 100 by using an external processor 211d.

20 **[0056]** The network storage controller 210d includes the first port PT1 connected with the Ethernet switch 100. The first port PT1 may be a hardware port for connecting the network storage controller 210d with a high-speed Ethernet system such as a fiber channel or InfiniBand. The first port PT1 may receive a command or data of the Ethernet protocol from the Ethernet switch 100. The first port PT1 transmits the received command or data to the host interface 215. Here, the embedded CPU 211c, the working memory 213, the host interface 215, the buffer manager 217, and

25 the flash interface 219 may be configured the same as those of FIG. 5.

[0057] In contrast, the network storage controller 210d may perform interfacing between the Ethernet switch 100 and the NVM cluster 280 by using the external processor 211d. The external processor 211d may translate a command or data of the Ethernet protocol transmitted from the Ethernet switch 100. In addition, the external processor 211d may control the network storage controller 210d so as to search for data stored in the NVM cluster 280 or to output the found data to the Ethernet switch 100 depending on a request provided from the Ethernet switch 100. The external processor 211d may perform distributed processing on a workload with the embedded CPU 211c. In an embodiment, the external processor 211d may process an assigned command under control of the embedded CPU 211c. In addition, the external processor 211d may perform processing of a separate operation requested from a host, such as network management. The external processor 211d may include one or more processors. The fourth port PT4 for connecting the external processor 211d and the network storage controller 210d is provided. The external processor 211d and the network storage controller 210d may communicate with each other through the fourth port PT4, for example, supporting a PCIe interface.

[0058] FIG. 7 is a diagram illustrating a part of an NVM cluster of the inventive concept. Referring to FIG. 7, the NVM array 240 may be directly connected with the network storage controller 210 and may exchange data.

[0059] The NVM array 240 may include, for example, NAND flash memory devices Raw NAND₁₁, Raw NAND₁₂, Raw NAND₁₃, ..., Raw NAND_{1i} (i being a natural number greater than 3). Each of the NAND flash memory devices Raw NAND₁₁, Raw NAND₁₂, Raw NAND₁₃, ..., Raw NAND_{1i} may not include a separate interface or switch between the first channel CH1 and a device. The NAND flash memory devices Raw NAND₁₁, Raw NAND₁₂, Raw NAND₁₃, ..., Raw

NAND_{1i} may be a combination of single devices or may be provided in the form of one package in which a plurality of devices are included.

[0060] Each of the NAND flash memory devices Raw NAND₁₁, Raw NAND₁₂, Raw NAND₁₃, ..., Raw NAND_{1i} is connected with the first channel CH1. For example, the first channel CH1 may be a data path between the network storage controller 210 and the NAND flash memory devices Raw NAND₁₁, Raw NAND₁₂, Raw NAND₁₃, ..., Raw NAND_{1i} sharing the same data line DQ. That is, the NAND flash memory devices Raw NAND₁₁, Raw NAND₁₂, Raw NAND₁₃, ..., Raw NAND_{1i} connected with the first channel CH1 may share the same data line DQ. Through the first channel CH1, the flash interface 219 of the network storage controller 210 may select any one of the NAND flash memory devices Raw NAND₁₁, Raw NAND₁₂, Raw NAND₁₃, ..., Raw NAND_{1i} or may control channel-way interleaving.

[0061] FIG. 8 is a diagram illustrating another part of an NVM cluster of the inventive concept. Referring to FIG. 8, the NVM array 250 may be connected with the network storage controller 210 through the NVM switch 230.

[0062] The NVM switch 230 communicates with the network storage controller 210 through the second channel CH2. For example, the NVM switch 230 may be connected with the flash interface 219 (refer to FIG. 2) of the network storage controller 210 through the second channel CH2. The flash interface 219 may access the NVM switch 230 and the NVM array 250 by using the second channel CH2 which is independent of the first channel CH1.

[0063] In an embodiment, the NVM switch 230 may include a switch controller 231 and a buffer 233. The switch controller 231 may perform an address mapping function for mapping an address provided through the second channel CH2

onto an actual address of the NVM array 250. The switch controller 231 may map an address of the NVM array 250 connected with a memory channel CH2_1 onto an address which the flash interface 219 provides. The switch controller 231 may also update address mapping when a device is added to or removed from the NVM array

5 250. The buffer 233 may temporarily store data exchanged between the network storage controller 210 and the NVM array 250. The buffer 233 may be implemented with, for example, a volatile memory such as a DRAM.

[0064] The NVM array 250 may include, for example, NAND flash memory devices Raw NAND_21, Raw NAND_22, Raw NAND_23, ..., Raw NAND_2j (i

10 being a natural number greater than 3). Each of the NAND flash memory devices Raw NAND_21, Raw NAND_22, Raw NAND_23, ..., Raw NAND_2j may be selected under control of the switch controller 231. The NAND flash memory devices Raw NAND_21, Raw NAND_22, Raw NAND_23, ..., Raw NAND_2j may be a combination of single devices or may be provided in the form of one package in which

15 a plurality of devices are included.

[0065] Each of the NAND flash memory devices Raw NAND_21, Raw NAND_22, Raw NAND_23, ..., Raw NAND_2j is connected with the memory channel CH2_1. For example, the memory channel CH2_1 may be a data path between the NVM switch 230 and the NAND flash memory devices Raw NAND_21, Raw

20 NAND_22, Raw NAND_23, ..., Raw NAND_2j sharing the same data line DQ. That is, the NAND flash memory devices Raw NAND_21, Raw NAND_22, Raw NAND_23, ..., Raw NAND_2j connected with the memory channel CH2_1 may share the same data line DQ.

[0066] An example in which the NAND flash memory devices Raw

25 NAND_21, Raw NAND_22, Raw NAND_23, ..., Raw NAND_2j are connected with

the network storage controller 210 through the NVM switch 230 is described above. Here, a description is given as one memory channel CH2_1 is provided in the NVM switch 230. However, it may be understood that the inventive concept is not limited thereto. That is, it may be understood that two or more memory channels may be connected with the NVM switch 230 and a plurality of NVM arrays are connected with the memory channels.

[0067] There is a substantial limitation on expansion of the SSD through a PCIe switch. However, theoretically, there is no limitation in expanding the NAND flash memory devices Raw NAND_21, Raw NAND_22, Raw NAND_23, ..., Raw NAND_2j of a device unit through the NVM switch 230. Accordingly, in the case of using the technology of the inventive concept, the capacity of the network storage device 200 may be easily expanded.

[0068] FIG. 9 is a diagram illustrating another part of an NVM cluster of the inventive concept. Referring to FIG. 9, the NVM arrays 260 and 270 may be connected with the network storage controller 210 through the NVM switches 232 and 234. Here, the NVM switches 232 and 234 may have a cascade connection structure.

[0069] The NVM switch 232 is connected with the network storage controller 210 through the third channel CH3. The NVM switch 232 may be connected with the flash interface 219 (refer to FIG. 3) of the network storage controller 210 through the third channel CH3. The flash interface 219 may access the NVM switch 232 by using the third channel CH3 which is independent of the first channel CH1 and the second channel CH2.

[0070] In an embodiment, the NVM switch 232 may include a switch controller 235 and a buffer 237. The switch controller 235 may perform an address mapping function for mapping an address provided through the third channel CH3

onto addresses of the NVM array 270 and the NVM switch 234 connected in the cascade form. The switch controller 235 may map an address of the NVM array 270 connected with a memory channel CH3_1 onto an address which the flash interface 219 provides. The switch controller 235 may map an address of the NVM switch 234
5 connected with a memory channel CH3_2 onto an address which the flash interface 219 provides.

[0071] The NVM switch 234 is connected with the NVM switch 232 through the memory channel CH3_2. The NVM switch 234 may be connected with the NVM array 260 through a memory channel CH3_3. The NVM switch 234 may include a
10 switch controller 236 and a buffer 238.

[0072] The switch controller 235 or the switch controller 236 may recognize an additional NVM array or NVM switch through an added memory port (memory channel) and may perform interfacing. Here, the NAND flash memory devices Raw NAND_31, Raw NAND_32, Raw NAND_33, ..., Raw NAND_3k and Raw
15 NAND_41, Raw NAND_42, Raw NAND_43, ..., Raw NAND_4m constituting the NVM arrays 260 and 270 may be a combination of single devices or may be provided in the form of a package including a plurality of devices.

[0073] An example in which the NVM arrays 240, 250, 260, and 270 are connected with the network storage controller 210 through the NVM switches 230, 232, and 234 are described above. Here, the NVM switches 230, 232, and 234 may be
20 connected in the cascade manner, and may be connected with a single NVM product or an array unit, not an SSD unit. There is a limitation on expansion of the SSD through a PCIe switch. However, theoretically, there is no limitation in expanding a storage capacity in a device unit through the NVM switches 230, 232, and 234. Accordingly,

in the case of using the technology of the inventive concept, the capacity of the network storage device 200 may be easily expanded.

[0074] FIG. 10 is a block diagram illustrating a network storage system configurable according to an embodiment of the inventive concept. Referring to FIG. 10, a network storage device 300 may include a network storage controller 310 for performing interfacing with the Ethernet switch 100, an external processor 305, a buffer memory 330, and a NVM cluster 380. The external processor 305 may be implemented with a processor or a computing device provided for a specific purpose, or an acceleration engine.

10 **[0075]** A host (not illustrated) such as a server or a data center transmits a command and data of the Ethernet protocol to the network storage device 300 through the Ethernet switch 100. The network storage device 300 may translate the command and data of the Ethernet protocol into a command and data to be transmitted to the NVM cluster 380 without translation into a PCIe protocol.

15 **[0076]** The network storage controller 310 controls the direct NVM cluster 380 in response to the command or the data of the Ethernet protocol provided through the Ethernet switch 100. In particular, the network storage controller 310 may be controlled by the external processor 305. It may be understood that the external processor 305 may be implemented with one or more processors for the purpose of exchanging data with the high-capacity NVM cluster 380 and processing data or a command. The external processor 305 may be implemented with an embedded CPU within the network storage controller 310.

[0077] In the network storage controller 310, transmission formats for supporting the Ethernet protocol, the NVMe-oF protocol and the NVMe protocol may be processed at a single layer. The network storage controller 310 provides interfacing

25

between the Ethernet switch 100 and the network storage device 300. The network storage controller 310 may directly translate a command or data of the Ethernet protocol format (e.g., a packet) provided from the Ethernet switch 100 to a command or data format to be applied to the NVM cluster 380.

5 **[0078]** The buffer memory 330 may be used as a data buffer for data exchange between the network storage device 300 and the Ethernet switch 100. For example, the buffer memory 330 may be implemented with a DRAM. However, the buffer memory 330 is not limited to the DRAM. For example, the buffer memory 330 may be implemented with a volatile memory, such as an SRAM or an SDRAM, a nonvolatile
10 RAM NV-RAM, or a combination thereof.

[0079] The NVM cluster 380 may include one or more NVM switches 330, 332, 334, 326, and 338 and one or more NVM arrays 340, 350, 352, 354, 360, 362, 370, 372, and 374. Here, each of the NVM arrays 340, 350, 352, 354, 360, 362, 370, 372, and 374 may be implemented with an array of nonvolatile memory devices. That
15 is, each of the NVM arrays 340, 350, 352, 354, 360, 362, 370, 372, and 374 may include a plurality of NAND flash memory devices.

[0080] The NVM switches 330, 332, 334, 326, and 338 are provided as a configuration for expansion-connecting a nonvolatile memory device to a flash interface of the network storage controller 310. For example, each of the NVM
20 switches 330, 332, 334, 326, and 338 may independently perform address mapping on a nonvolatile memory device or a nonvolatile memory device array, which is connected thereto.

[0081] The NVM switch 330 may be connected with the three NVM arrays 350, 352, and 354 and may perform interfacing with the network storage controller
25 310. The NVM switch 332 may perform interfacing between the NVM switch 334 and

the network storage controller 310. The NVM switch 334 may control the NVM switch 336 and the NVM array 360. The NVM switch 336 may control the NVM switch 338 and the NVM array 362. The NVM switch 338 may be connected with the three NVM arrays 370, 372, and 374 and may perform interfacing with the NVM switch 336.

5 **[0082]** A connection form of the NVM switches 330, 332, 334, 326, and 338, which may be connected with the network storage controller 310, and the NVM arrays 340, 350, 352, 354, 360, 362, 370, 372, and 374 is briefly described above. Of course, it may be well understood that the connection with the network storage controller 310 may be made without an NVM switch like the NVM array 340.

10 **[0083]** FIG. 11 is a block diagram illustrating a network storage system according to an embodiment of the inventive concept. Referring to FIG. 11, a network storage system 1000 of the inventive concept may include a server 1100, a network fabric 1200, and a plurality of Ethernet SSDs 1300 and 1400.

15 **[0084]** The server 1100 is connected with the plurality of Ethernet SSDs 1300 and 1400 through the network fabric 1200. The server 1100 may transmit a command and data to the plurality of Ethernet SSDs 1300 and 1400 by using the Ethernet protocol. The server 1100 may receive data of the Ethernet protocol provided from at least one of the plurality of Ethernet SSDs 1300 and 1400. The network fabric 1200 may be a network switch.

20 **[0085]** Each of the plurality of Ethernet SSDs 1300 and 1400 may be implemented with a storage device of FIG. 1 or 10. That is, like the Ethernet SSD 1300, a network storage controller 1310 may be directly connected with an NVM array 1320 without passing through an NVM switch. In contrast, a network storage controller 1410 included in the Ethernet SSD 1400 may control NVM arrays 1422 and
25 1424 through NVM switches 1412 and 1414. Of course, it may be understood that the

Ethernet SSD controller 1410 may control the NVM array 1420 directly without using an NVM switch.

[0086] According to embodiments of the inventive concept, a network storage device, the capacity of which may be expanded in units of nonvolatile memory devices or nonvolatile memory arrays such as a NAND flash memory device, may be provided. Accordingly, the network storage device of the inventive concept may be connected with a network fabric to perform protocol translation of a single layer with respect to multiple protocols, thus reducing a latency and expanding a memory capacity easily.

[0087] While the inventive concept has been described with reference to exemplary embodiments thereof, it will be apparent to those of ordinary skill in the art that various changes and modifications may be made thereto without departing from the spirit and scope of the inventive concept as set forth in the following claims.

WHAT IS CLAIMED IS:

1. A network storage device connected with a network fabric, comprising:
a network storage controller configured to perform interfacing with the
5 network fabric and to directly translate and execute a first command provided through
the network fabric into a second command for controlling a nonvolatile memory
cluster, and implemented with a single chip; and
the nonvolatile memory cluster configured to exchange data with the network
storage controller under control of the network storage controller,
10 wherein the nonvolatile memory cluster includes:
a first nonvolatile memory array directly connected with the network storage
controller through a first channel;
a nonvolatile memory switch connected with the network storage controller
through a second channel; and
15 a second nonvolatile memory array configured to communicate with the
network storage controller under control of the nonvolatile memory switch.
2. The network storage device of claim 1, wherein the first command
includes a network transmission format.
20
3. The network storage device of claim 1, wherein the network storage
controller performs a command for the nonvolatile memory cluster depending on the
second command.
- 25 4. The network storage device of claim 3, further comprising:

a buffer memory,

wherein the network storage controller further includes:

a first port for a communication with the network fabric;

a second port for a communication with the buffer memory; and

5 a third port for a communication with the nonvolatile memory cluster.

5. The network storage device of claim 4, wherein the first port includes an interface which supports a network transmission format.

10 6. The network storage device of claim 2 or claim 5, wherein the network transmission format includes a high-speed Ethernet system.

7. The network storage device of claim 1, wherein the network storage controller includes:

15 a storage controller configured to translate the first command provided from the network fabric as a command into the second command and to control the nonvolatile memory cluster depending on the second command; and

an external computing device configured to perform a command for the nonvolatile memory cluster depending on a request of the storage controller.

20

8. The network storage device of claim 7, wherein the external computing device includes one or more processors, which perform distributed processing based on the first command or the second command with the storage controller or solely perform processing based on the first command or the second command, or an

25 acceleration engine.

9. The network storage device of claim 1, wherein the network storage controller includes:

5 a host interface configured to receive the first command provided from the network fabric as a command;

a storage controller configured to translate the first command into the second command for controlling the nonvolatile memory cluster and to perform a command for the nonvolatile memory cluster depending on the second command; and

10 an interface configured to access the nonvolatile memory cluster under control of the storage controller.

10. A network storage device comprising:

15 a network storage controller configured to directly translate a first command provided through a network switch into a second command for controlling a nonvolatile memory and to output the second command by using an interface, and implemented with a single chip; and

a nonvolatile memory cluster configured to store write data depending on the second command output from the network storage controller or to output stored data to the network storage controller,

20 wherein the network storage controller includes:

one or more external computing devices configured to perform a command for the nonvolatile memory cluster based on the second command,

wherein the nonvolatile memory cluster includes:

25 a first nonvolatile memory array directly connected with the network storage controller through a first channel and controlled through the interface.

11. The network storage device of claim 10, wherein the first command has a network transmission format, and the second command has a format for controlling the nonvolatile memory.

5

12. The network storage device of claim 10 or claim 11, wherein the one or more external computing devices include one or more processors which process a command according to a request of a host.

10 13. The network storage device of claim 12, wherein the one or more external computing devices perform a network management operation according to the request of the host.

14. The network storage device of claim 10, wherein the network storage
15 controller includes a storage controller including an embedded computing device, and wherein the one or more external computing devices include one or more processors which perform distributed processing on the first command or the second command with the embedded computing device.

20 15. The network storage device of any one of claims 10 to 14, wherein the nonvolatile memory cluster further includes a second nonvolatile memory array, and further comprising:

a first nonvolatile memory switch connected with the network storage
controller through a second channel and configured to perform interfacing between the
25 second nonvolatile memory array and the interface.

16. The network storage device of claim 15, wherein the first nonvolatile memory switch includes:

a first switch controller configured to map a first address provided from the
5 interface onto the second nonvolatile memory array; and

a first buffer configured to buffer data transmitted between the interface and the second nonvolatile memory array.

17. The network storage device of claim 16, wherein the nonvolatile
10 memory cluster further includes:

a third nonvolatile memory array; and

a second nonvolatile memory switch configured to perform interfacing between the first nonvolatile memory switch and the third nonvolatile memory array.

15 18. The network storage device of claim 17, wherein the first and second nonvolatile memory switches perform address mapping independently of each other.