



- (51) International Patent Classification:  
*G06F 12/00* (2006.01)    *G06F 13/14* (2006.01)
- (21) International Application Number:  
PCT/US2014/011352
- (22) International Filing Date:  
13 January 2014 (13.01.2014)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:  
13/741,299    14 January 2013 (14.01.2013)    US
- (71) Applicant: WESTERN DIGITAL TECHNOLOGIES, INC. [US/US]; 3355 Michelson Drive, Suite 100, Irvine, California 92612 (US).
- (72) Inventors: JEAN, Sebastien A.; 97 Alhambra, Irvine, California 92620 (US). HORN, Robert L.; 3355 Michelson Drive, Suite 100, Irvine, California 92612 (US).
- (74) Agent: CHAN, David H.; Western Digital Technologies, Inc, 3355 Michelson Drive, Irvine, California 92612 (US).
- (81) Designated States (unless otherwise indicated, for every kind of national protection available): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT,

HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, 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, 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))

**Published:**

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

(54) Title: SYSTEMS AND METHODS OF CONFIGURING A MODE OF OPERATION IN A SOLID-STATE MEMORY

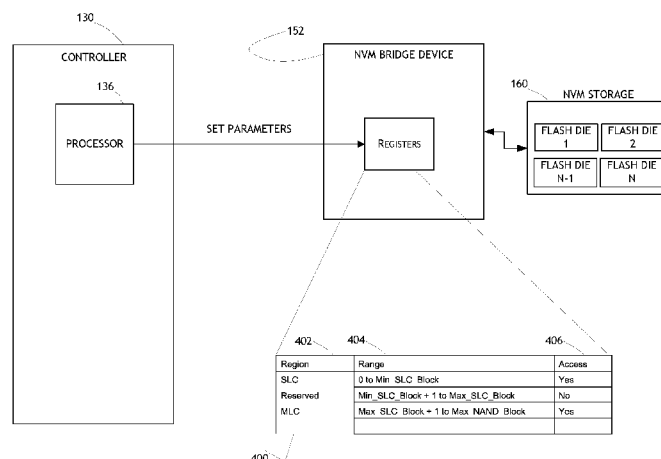
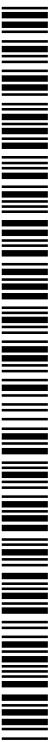


FIGURE 4

(57) Abstract: Disclosed herein is an architecture that pairs a controller with a NVM (non-volatile memory) storage system. The NVM storage system includes a bridge device that communicates with the controller. In one embodiment, the bridge device allows for certain data locations (blocks, pages or units at any other granularity) in the flash dies to be (1) placed into a reserved mode where data access is prevented (2) assigned into an SLC (Single-Level Cell) mode or an MLC (Multi-Level Cell) mode in response to controller command, (3) made available for data access after the assignment of mode. This flexibility enables the controller to increase SLC mode or MLC mode data locations based on run-time conditions. In one embodiment, the assignment of the reserved data locations is performed in a way to ensure that warranty conditions imposed by the memory vendors are observed.



## SYSTEMS AND METHODS OF CONFIGURING A MODE OF OPERATION IN A SOLID-STATE MEMORY

### BACKGROUND

#### Technical Field

[0001] This disclosure relates to non-volatile storage systems, including but not limited to flash drives. More particularly, the disclosure relates to systems and methods for an enhanced controller architecture in solid state drives.

#### Description of the Related Art

[0002] Various types of controller architecture exist for controlling flash media. The Open NAND Flash Interface (ONFI) is a standard interface that specifies some common sets of commands that flash memory manufacturers should support. ONFI supports some low level rudimentary I/O operations that can include, for example, page write/read and block erase. However, effective flash media management often involves a number of high level and potentially process-intensive functions such as logical-to-physical mapping, garbage collection, and wear leveling. These functions are beyond the scope of ONFI and thus an effective controller architecture needs to address these needs while providing a high level of data throughput performance to the host.

### BRIEF DESCRIPTION OF THE DRAWINGS

[0003] Systems and methods that embody the various features of the invention will now be described with reference to the following drawings, in which:

[0004] Figures 1A-1C illustrate several solid-state drive (SSD) controller architectures.

[0005] Figure 1D is a block diagram showing the controller architecture according to an embodiment.

[0006] Figures 2A and 2B are block diagrams showing the controller architecture according to some embodiments.

[0007] Figure 3 is a block diagram illustrating the command processing components between the controller and the bridge device according to one embodiment.

[0008] Figure 4 is a block diagram showing how the bridge device responds to mode configuration commands from the controller according to one embodiment.

**[0009]** Figure 5A and 5B illustrate example operating mode configuration settings of data locations according to one embodiment.

**[0010]** Figure 6 is a flow diagram showing a process of configuring operating mode of data locations according to one embodiment.

**[0011]** Figure 7A and 7B show example operating mode configuration settings for multiple ranges of data locations according to various embodiments.

**[0012]** Figures 8A and 8B show example configurations across multiple memory devices according to various embodiments.

## **DETAILED DESCRIPTION**

**[0013]** While certain embodiments of the inventions are described, these embodiments are presented by way of example only, and are not intended to limit the scope of the inventions. Indeed, the novel methods and systems described herein may be embodied in a variety of other forms. Furthermore, various omissions, substitutions and changes in the form of the methods and systems described herein may be made without departing from the spirit of the inventions.

### **I. Controller/Bridge Designs**

**[0014]** Figures 1A-1C illustrate a number of SSD controller architectures. Figure 1A illustrates a SSD controller that includes an ONFI (Open NAND Flash Interface) interface. The ONFI interface is a low level, parallel I/O interface that provides basic command support to enable an external component such as a host controller to control operations in a NAND. Figure 1A shows a typical setup in which a host device such as a computer that includes a SSD controller, which in turn includes an ONFI interface for controlling a non-volatile memory (NVM) unit that supports ONFI. The ONFI interface provides some basic level of control over operations such as page program, page read, block erase, etc. Current versions of ONFI support one command per die and plane, and provide some basic queuing capability through commands such as “read multiple” and “write multiple” commands. However, there can be no mixing of different types of commands. Moreover, commands are batch processed, which means that the commands must clear the queue completely before more commands can be accepted. The SSD controller in this approach

would perform high level NVM management functions such as garbage collection, wear leveling, mapping etc.

**[0015]** Figure 1B illustrates an alternative approach in which the NVM is coupled with a bridge controller/device that performs some level of basic channel management of the NAND and signal processing. The bridge may provide an ONFI or ONFI-equivalent interface to the SSD controller. However, this interface may be modified from the standard ONFI interface, and may support additional capabilities such as supporting multiple commands. The SSD controller, in this design, would still perform high level NAND management functions such as garbage collection, and communicate with the bridge via the ONFI or ONFI-equivalent interface.

**[0016]** Figure 1C depicts a third approach in which the NVM is coupled with a bridge in a storage system. In this approach, the bridge performs a number of high level NVM management functions such as garbage collection, wear leveling, and mapping, as well as the lower level functions such as channel management and signal processing. The storage system is coupled with the host device through a high level I/O interface such as eMMC or UFS. This is a common design found in many memory card products. The host sends over commands to the storage system (such as read and write commands) using logical addressing. Features such as advanced command queuing, health reporting, and detailed error reporting may be supported.

## **II. System Overview**

**[0017]** Figure 1D is a block diagram that illustrates a controller design architecture according to some embodiments of the invention. Figure 1D shows a SSD controller performing high level NVM management functions such as garbage collection, wear leveling, etc. In one embodiment, the SSD controller is coupled with a NVM storage system over a high level, high speed interface such as PCIe and eMMC. In lieu of PCIe or eMMC, other standardized and/or proprietary interfaces could be extended for use as this bus interface. The NVM storage system in one embodiment includes a bridge that communicates with the SSD controller via the high level, high speed interface, and controls the NAND memory with a low level interface such as ONFI. As shown, additional features such as advanced command

queuing, health reporting, and detailed error reporting may be supported over the high level interface.

**[0018]** Unlike the designs described above, the controller in this architecture is provided a rich set of physical level of controls over individual elements of the NVM (e.g., page level control) over a sophisticated and fast interface such as PCIe. It is observed that in many controller-bridge type designs that the bridge is typically implemented on a processor with reduced performance due to power concerns, while the controller typically is in an environment that faces less power concerns. If the processor intensive functionalities are shifted to the higher performing controller, then overall latency can be reduced. Thus, in some embodiments, the controller is typically implemented in a higher powered processor that can support the advanced NVM management functions. The bridge, on the other hand, is implemented in some embodiments in a lower powered processor to minimize energy usage of the overall NVM storage module/unit. As a result, the bridge may perform basic signal processing and channel management of the NVM, as well as some basic error correction functions and XOR parity accumulation. In some embodiments, the controller performs logical-to-physical address mapping, garbage collection, wear leveling, parity management (via control of the parity accumulator in the bridge), RAID striping, etc. This division of labor still provides the controller direct, physical (e.g., page-level) control of the NVM, resulting in the controller managing the NVM at both the page and block level over a fast, high level interface such as PCIe. The controller in one embodiment also manages other integrated services such as the XOR parity accumulator in the bridge.

**[0019]** In one embodiment, another advantage of the architecture's division of the management tasks relates to NVM industry trends. Physical management of the NVM is becoming increasingly important as the most common type of NVM, e.g., MLC (Multi-Level Cell) NAND, continues to evolve to provide higher capacity at the cost of reduced endurance. For example, today's MLC products with a 5,000 P/E-cycle endurance are being replaced with next-generation MLC products with a 1,500-2,000 P/E-cycle endurance. Commonly, the bridge designer is in the best position to understand the physical properties of the NVM and how best to extend its life by implementing various endurance enhancement/management functionalities. Because of this rapidly changing technological landscape, and because each

individual NVM manufacturer may require different such endurance enhancement/management functionalities, these functionalities may constantly require fine-tuning to suit the diverse and ever-change varieties of NVM products. Thus, the architecture of some embodiments provides another advantage in its labor division by isolating these functions in the bridge and allowing the controller designer to focus on the high level data management functions. In other words, since the controller and the bridge have different design constraints and priorities, under the architecture each may be updated according to different schedules and manners without a complete redesign of the whole.

**[0020]** With the reduced latency provided by the design, the bridge may be paired with less expensive media. For example, the bridge may be paired with MLC NAND rather SLC (Single-Level Cell) NAND while still meeting performance metrics demanded by customers. In addition, in some embodiments the above described controller-bridge design can be adapted for use in a hybrid drive comprising flash memory and hard disk components. In those embodiments the controller, in addition to managing data accesses to the NVM through the bridge, would also manage data accesses to one or more hard drives. Additional features of this design will be further illustrated below with various drawings and descriptions of the embodiments of the invention.

## **II.A. Controller-Bridge Implementation**

**[0021]** Figure 2A shows an embodiment of a controller-bridge architecture previously introduced in Figure 1D. As shown, a solid-state non-volatile storage system 120 is connected to a host system 110. The host system 110 communicates with the non-volatile storage system 120 using a storage interface 112. The host's storage interface 112 can communicate with the non-volatile storage system 120 using any known communication protocol, such as SATA, SCSI, SAS, USB, Fibre Channel, PCIe, eMMC, etc.

**[0022]** In one embodiment, the non-volatile storage system 120 includes a controller 130 and a NVM storage module 150. The controller 130 in one embodiment communicates with a bridge device 152 within the NVM storage module 150 via a high level interface such as PCIe (through bus logic/interface 140). PCIe is used in one embodiment as it defines a rich packet based routing and Quality of

Service (QoS) infrastructure and provides a high speed interface. The controller may include a processor 136 to control data functions, and the core may be coupled with static memory 132 and dynamic memory 134. The controller 130 may also include a data path 138 for processing/transferring data related to data access commands from the host system 110. In one embodiment, the controller 130 is implemented on a SoC (System on Chip), though those skilled in the art will recognize that other hardware/firmware implementations are possible.

**[0023]** In one embodiment, the use of PCIe means that the address range assigned to a device function is used for packet routing both on the fabric and within the device. In one embodiment, the PCIe transaction layer delivers packets to an internal register interface that is read by firmware. Advanced devices often direct incoming packets to internal RAM or hardware acceleration modules.

**[0024]** The bridge device 152 in one embodiment comprises bus logic/interface 154 for communicating with the bus logic/interface 140 (on the controller 130) over the high level interface bus. On the other end of the bridge, the bridge device 152 includes a low level interface 158 such as ONFI for communicating with the NVM storage 160 (e.g., NAND), which may include a number of storage devices such as flash dies 162, 164, 166, and 168. Although ONFI is depicted in this embodiment, other suitable flash memory interfaces may be used. In another embodiment, the bridge may use a different interface such as Toggle or a proprietary interface to communicate with the NVM storage 160, or send direct commands to the storage.

## **II.B. Division of Labor**

**[0025]** The advantages of dividing the NVM management functions were outlined above in Section II. In particular, the architecture reduces latency and addresses the various design constraints while allowing the controller and bridge designers to optimize their respective portions of the architecture. In one embodiment, the controller is responsible for block level management, parity stripe layout, garbage collection, wear leveling, handling read disturb and error recovery. The bridge device, in one embodiment, manages the raw NVM flash interface. It may also provide one or more of: command queuing, error correction, XOR parity accumulator, data protection, and enhances block endurance. The interface between the bridge and the controller in one embodiment is a lightweight PCIe-

based data and management interface. The controller uses the interface control command to configure the bridge and data commands to access the NVM media.

**[0026]** It is also noted that the controller uses physical page addressing instead of a logical one that is common in existing controller-bridge designs. The bridge can identify the relationship between pages, blocks, planes, and dies. This gives the controller the greatest flexibility for creating RAID stripe layout, performing data moves, and handling bad blocks. These details are abstracted from the bridge. When using direct addressing, the controller simply provides a set of direct page addresses in the command header to the bridge. In one embodiment, the pages are not necessarily sequential or even in the same block. In most cases, the controller will access pages spread across multiple planes and multiple dies in order to maximize concurrent hardware access.

### **II.C. Hybrid Application**

**[0027]** Some embodiments of the controller-bridge architecture can be adapted for other uses. For example, Figure 2B shows the use of the controller architecture in a hybrid drive 122, which in addition to the NVM and bridge components described above, includes a magnetic storage module 180 which has a magnetic media 184 such as a rotating hard disk drive (HDD). The controller 130 in this embodiment would thus manage data accesses to both the NVM storage module 150 and magnetic storage module 180. In one embodiment, a different interface than interface 140 (which connects to the NVM) may be used to connect the controller 130 to the magnetic storage module 180.

**[0028]** The hybrid application illustrates an additional advantage of the controller architecture. A hybrid drive typically includes an SSD that has its own internal controller that has a mapping table to address the NVM within the SSD. While the HDD portion of the hybrid is typically addressed directly, the hybrid controller uses a special mapping table to determine whether data is in the SSD or the HDD. The use of this special mapping table along with the internal SSD mapping table introduces duplicate overhead in cases where data is accessed in the SSD portion of the hybrid, since there are two mapping tables and there is significant cost associated with maintaining each table.



**[0029]** In contrast, because the controller 130 in the present architecture manages the NVM at both the block and page levels and the magnetic media, it can provide uniformed address management across the flash and magnetic media in a single location. Hence there is no need to have the two tables noted above. This has the advantages of reducing duplicative table lookups and all the associated costs/complexities related to maintaining separate mapping tables. Direct page addressing is used in the unified mapping scheme.

**[0030]** In addition, in a hybrid application, the NVM could still provide effective performance enhancement even when it has a large amount of bad blocks (e.g., 50%). In one hybrid embodiment, the controller also has efficient address gap handling capability (on gaps caused by the bad blocks). In an alternative hybrid embodiment, the unified addressing scheme does not necessarily require a bridge to work with the controller. The controller can potentially use a raw NVM interface (e.g., ONFI) to access the NVM.

### **III. Data Command Processing**

**[0031]** Figure 3 is a block diagram illustrating the command processing components between the controller and the bridge device according to one embodiment. Using the construct of the PCIe interface (or other equivalent interfaces), both the controller and the bridge implement its own address spaces (210, 250) in their respective device memories that can be addressed by the other device. Messages are communicated by writing to queues located within certain addresses within the address spaces, and the addresses are stored in the configuration and status registers 252 in one embodiment. The use of individual queues to handle data access commands and communications between the controller and the bridge will be further described below.

#### **III.A. Command and Admin Queues - Bridge**

**[0032]** In one embodiment, the controller sends data access commands to a command queue 262 in the bridge device. This is performed by the controller sending data command messages to the bridge (by writing to the command queue BAR (base address register)). In one embodiment, the command queue has room for sixteen messages, though the number of messages can vary in other

embodiments. The command queue can be implemented in a number of ways. One option is full hardware automation where the controller simply writes to a fixed offset. Alternately it can be implemented in memory using a ring buffer or an array based linked list. In one embodiment, the implementation must allow for efficient insertion and notification with minimal bus traffic. In one embodiment, the controller knows the current queue depth based on the number of status responses the bridge has sent back (e.g., a message to the controller's completion queue indicating completion of a command). Note that the data commands are much smaller than the actual data. A given record slot in the queue is considered available once the bridge sends back a completion status or an error report.

**[0033]** In the embodiment shown in Figure 3, the bridge side 250 also implements configuration and status registers (CSR) 252, along with an admin queue 258 for receiving, from the controller, command messages related to operation of the command queue (e.g., a message for pausing the command queue) or administrative messages concerning the bridge's operation in general. The admin queue 258 may be implemented in a fashion that is similar to the command queue, such as through full hardware automation or ring buffer. Also, like the command queue, the admin queue may be configured for efficient insertion and notification with minimal bus traffic. Like the command queue, the controller can derive the current queue depth and available slots based on responses from the bridge.

### **III.B. Status Queues - Controller**

**[0034]** On the controller side 210 are a set of data ports 214 at data addresses 212 and a number of status queues. In one embodiment, the status queues include an error queue 218, an information queue 222, and a command completion queue 226. These queues are responsible for receiving messages from the bridge regarding command processing, as well as current status of the bridge and the NVM.

### **III.C. Communications Between the Controller and the Bridge**

**[0035]** Communications between the controller and the bridge are effected in one embodiment through a PCIe protocol stack 230 which includes a number of layers on both sides, including a transactional layer (232, 242), a data link layer (234, 240),

and physical layer (236, 238). While PCIe is used in this disclosure to illustrate the operation of the controller and the bridge, other similar standards can be used as well.

**[0036]** The PCIe transaction layer assigns transmit credits based on how much room is left in its Virtual Channel (VC) buffer space. According to the PCIe specification, devices must implement VC0, though some devices implement additional VC to ensure high priority messages have dedicated resources. Packets are directed to the appropriated VC based on their Traffic Class (TC). The TC is also used to determine priority when packets are flowing over the PCIe fabric. Higher TC packets are generally given priority by the root complex, switches and end-devices.

**[0037]** In one embodiment, the controller is designed to operate using only VC0. In one embodiment, though the bridge may implement additional VC, it must be configurable so that it can operate in single VC mode. The messages communicated between the controller and the bridge will be better understood in view of the following brief description of the data processing flow. To service a read command from the host, the controller may first send a command message to the bridge's command queue. Once the bridge processes the command message, it will read the requested data from the NVM and send the read data back to a corresponding data port on the controller side. This action triggers the data path on the controller, which leads to the data being sent back to the host. Conversely, to service a write command from the host, the controller may first send a command message to the bridge's command queue. Once the bridge processes the command message, it will read from a corresponding data port on the controller side. This action triggers the data path on the controller, which leads to the write data being sent from a buffer in the controller to the bridge for writing to the NVM.

**[0038]** The controller in one embodiment communicates with the bridge using three message types of increasing priority: data to be written to the NVM for write commands (0), messages for the bridge's command queue (1), and messages for the bridge's admin queue (2). Those skilled in the art will recognize that different priorities may be assigned to these messages, and the messages could be combined into fewer types or divided into more types depending on the

implementation. In one embodiment, under normal conditions, the controller sends a steady stream of data packets to the bridge.

**[0039]** In one embodiment, the bridge interacts with the controller using its own set of prioritized message types (listed here in increasing priority): data read from the NVM for read commands (0), messages for the controller's completion/info queues (1), and messages for the controller's error queue (2). Those skilled in the art will recognize that different priorities may be assigned to these messages, and the messages could be combined into fewer types or divided into more types depending on the implementation. As will be further described below, to facilitate fast processing of data access commands, a read or write by the bridge to a data port in the controller automatically triggers the data path in the controller. In one embodiment, it is not uncommon for the bridge to process several commands in parallel. In one embodiment, the bridge uses the completion queue 226 to notify the controller when commands have completed successfully. In addition, non-critical messages are sent to the info queue 222 while detailed error reports are sent to the error queue 218. In other embodiments, these queues may be combined into fewer queues (with different message types being distinguished by special flags or implied address values) or separated into more queues (e.g., different error queues for different types of error or different info queues for different types of information returned from the bridge).

**[0040]** In other embodiments where a different interface than PCIe is used, the PCIe protocol stack may be replaced with the appropriate stacks/layers of that interface. Those skilled in the art will recognize that other equivalent standardized interfaces (e.g., eMMC) may be adapted to replace PCIe. In other embodiments, a custom/proprietary interface may be used to handle communications between the controller and the bridge.

#### **IV. SLC/MLC Mode Management**

**[0041]** A common type of solid state memory used is Multi-Level Cell (MLC) NAND memory, which is ordinarily capable of operating in at least one Multi-Level Cell (MLC) mode or a Single-Level Cell (SLC) mode. In one embodiment, the NVM Bridge architecture described above allows for certain data locations (blocks, pages, or units at any other granularity) in the flash dies to be (1) placed into a reserved

mode where data access is prevented, (2) assigned into an SLC mode or an MLC mode in response to controller command, (3) made available for data access after the assignment of mode. This flexibility enhances the functionality of the overall system in several ways, including giving the controller a way to increase SLC mode or MLC mode data locations based on run-time conditions and/or product configurations. In one embodiment, the assignment of the reserved data locations is performed in a way to ensure that external constraints, such as warranty conditions imposed by the memory vendors, are observed.

#### **IV.A. Overview**

**[0042]** Figure 4 is a block diagram showing how the bridge device responds to mode configuration commands from the controller according to one embodiment. As previously described above, in one embodiment, the bridge device 152 is coupled with a NVM storage 160 which includes one or more flash dies. In one embodiment, the bridge device takes commands from the controller 130 over a bus (e.g., a PCIe bus described above) and executes data access commands in the NVM storage 160. The type of bus protocol is not limited to PCIe and other protocols may be used in other embodiments. The bridge device 152 may also have some intelligence on how to extend the life of the flash dies.

**[0043]** The flash dies may include MLC cells that can operate in an MLC or an SLC mode. For example, in MLC cells that can store up to 2 bits of data per cell, placing a block in SLC mode reduces the data capacity from 2 bits per cell down to 1 bit per cell. On the other hand, operating in SLC mode yields better program/erase (P/E) endurance. It is common to have the MLC cells set to operate in one mode or another before the storage device reaches the field of use, as a result of warranty restrictions imposed by the memory manufacturer. In this disclosure, the term MLC refers to memory cells that are capable of storing 2 or more bits per cell. In implementations with more than 2 bits per cell, there exists several levels of MLC mode, and the adjustment methods disclosed here are applicable to assign cells to operate at those different MLC levels. For example, in memory where a 3-bit per cell mode is available, reserved region may be assigned to operate in a 3-bit per cell MLC mode, a 2-bit per cell MLC mode, or an SLC mode.

**[0044]** Some embodiments of the invention allow certain memory blocks to be placed, as needed, in a reserved region to be used by the controller 130 for improving endurance, performance and reliability at the cost of reducing storage capacity. Some examples may include managing high-access hotspots and storing critical system data. Commitment of the reserved blocks to either the MLC mode or the SLC mode does not need to be made at the beginning of life of the storage device and the controller has the flexibility to choose how much to allocate during the course of life. To achieve this, in one embodiment, the bridge device provides a method of allowing the controller to place a range of reserved data locations into SLC or MLC mode at run time. For example, the controller can set mode through sending one or more commands to the bridge device. For example, if PCIe is used, the CSR 252 as shown in Figure 3 could be used to allow the commands to be communicated through setting of fields inside the CSR.

**[0046]** In one embodiment, two register fields are used to establish the range of the reserved data locations: a starting address (e.g., `min_slc_block`) and an ending address (e.g., `max_slc_block`). The starting and ending addresses define the reserved region. In other embodiments, the controller 130 can indicate the range of the reserved region through one or more commands directly instead of using registers (e.g., sending commands over a bus with a protocol such as SATA).

**[0047]** As shown in the table 400 of Figure 4, in one embodiment, the starting and ending address values define three regions 402: SLC, Reserved, and MLC. Both the SLC and MLC regions can be accessed normally, as indicated by the access values 406. The Reserved region's access value indicates that it cannot be accessed to prevent the blocks within the region from being used. It is noted that table 400 is a teaching example intended to illustrate the concepts and that the actual values or types of values stored in the registers or sent by commands to achieve the same configuration may be different.

**[0048]** In one embodiment, the regions can be of a different configuration, as will be further shown in Figures 7A and 7B. In one embodiment, the configuration is applied to each plane across all dies. In another embodiment, each die or each subset of dies may have its own configuration of regions, as will be further shown in Figures 8A and 8B.

#### **IV.B. Procedure for Growing MLC or SLC Region**

**[0049]** Figure 5A and 5B illustrate example operating mode configuration settings of data locations according to one embodiment. Illustration 1 of Figure 5A shows an initial setting of an address range. The range is divided into a reserved region (0 to SLC\_Max) and an MLC region (SLC\_Max + 1 to NAND\_Max), where SLC\_Max and NAND\_Max are addresses in the range. For example, SLC\_Max may be 255 and NAND\_Max may be 1023 in a die with 1024 data blocks/data locations. On the left side of Illustration 1 are the boundary values of the Reserved Region, with a starting address (Min Block = 0) and an ending address (Max Block = SLC\_Max). As discussed above, the bridge device prevents access to data locations within the reserved region.

**[0050]** To change the size of the reserved region (and correspondingly, the size of an adjacent non-reserved region), the starting and ending address values may be changed via a command from the controller, or via setting of register values in the bridge device. In Illustration 2, the starting address (Min Block) is changed from 0 to 30. In Illustration 3, the Reserved region is adjusted correspondingly, so that it now starts at 30. The data locations in the range 0 to 29 are now assigned to an SLC region (i.e., assigned to operate in the SLC mode) and data access is now enabled. In other words, the SLC region has grown from 0 data locations to 30 data locations (0-29).

##### **IV.B.I Access Enablement Options**

**[0051]** The transition from Illustrations 2 to 3 can occur in a number of ways. In one embodiment, data access to the newly assigned data locations (e.g., 0-29 in Illustrations 2-3) occurs immediately upon receipt of a command from the controller to adjust the initial starting address to the new starting address. In other embodiments, data access to the newly assigned data locations can be delayed – in a direct or an indirect manner. In one embodiment, the access is enabled directly upon receipt of a later command from the controller to activate access. In another embodiment, the access is enabled indirectly upon receipt of a later command not specifically for access activation. For example, it could be a data access (e.g., write) command from the controller to one of the newly assigned data locations.

**[0052]** In Illustration 4, the ending address of the Reserved region is adjusted (Max Block is set to 80) and in Illustration 5 the MLC region grows to include those locations (80 – SLC\_Max) that were formerly in the Reserved region. Much like the transition in Illustrations 2 to 3, data access to those data locations that are newly assigned to operate in the MLC mode may be enabled immediately, in a delayed fashion through a direct or indirect mechanism, as described above.

#### **IV.B.II Growth Rules**

**[0053]** It is noted that in one embodiment the bridge device is configured to allow for repeated adjustment of the starting or ending address to assign data locations in the reserved region to either the SLC or MLC region. For example, in response to commands from the controller, the bridge device may successively adjust to a new starting address that is greater than a prior starting address (e.g., Illustrations 2 to 3) and after each successive adjustment, assign the data locations located between the new starting address and the prior starting address to operate in a mode to match the operating mode of the data locations in the region that are adjacent to the prior starting address (SLC region in this example). The same repetition can be done on the MLC region side / ending address side. Note that the layout in these illustrations may be reversed, so that an MLC region may be in the lower address range and the SLC region may be in the higher address range.

**[0054]** In one embodiment, the bridge device ensures the adjustments are made only in the direction of reducing the size of the reserved region. In other words, the data locations in the reserved region can be assigned out to operate in the SLC or MLC mode to accommodate the growth of the adjacent SLC or MLC region, but data locations from the adjacent SLC or MLC region are not re-designated as being in the reserved region. This procedure allows the memory to be used in observance with rules set forth by some manufacturers such as warranty agreements, which prohibit data locations from being reset into a different mode of operating once a mode has been chosen and the data locations have been used in the chosen mode.

#### **IV.B.III Reset Option**

**[0055]** In one embodiment, the bridge device does optionally allow such reset of the operating mode to occur through a reset function that is executed, for example,



based on a command from the controller 130. As shown in Illustration 6 of Figure 5B, the boundaries of the Reserved region are reset. Thus, data locations that were previously assigned to join the adjacent regions are now back in the Reserved region. For example, data location 30 that was previously shown in Illustration 3 to have been set to operate in the SLC mode is now free to be re-assigned to operate in the MLC mode.

**[0056]** Optionally, a reset flag may be set to indicate that such a reset has occurred. Setting this flag would indicate that an operation constraint has been voided such as the endurance warranty. In some instances, the reset function may be useful for testing / development purposes but may be disabled in the field of use.

#### **IV.B.IV Growth Summary, Example Use Cases**

**[0057]** The adjustment process is summarized in the flow chart of Figure 6. The process 500 shown may be executed by the controller and/or the bridge device. In block 504, the bridge device and/or the controller performs the initial setup of the reserved region (e.g., setting the boundaries, etc.). In block 506 and 508, data locations in the reserved region may be assigned to operate in either SLC or MLC mode, to facilitate the growth of adjacent region of that mode as shown in the example in Figures 5A and 5B. As shown in the flow chart, either assignment can be repeated multiple times, in consecutive order or in a mixed sequence. Optionally, a reset may be performed in block 510 and a reset flag may be set in block 512.

**[0058]** In one embodiment, the growth of the MLC or SLC region into the reserved region is constrained by a threshold. So the bridge device may be configured to not adjust to the new starting address when it exceeds a threshold set in accordance to a manufacturer warranty constraint.

**[0059]** As solid state memory dies in the same storage device differ in aspects such as endurance and quality (e.g., read errors encountered), the reserved regions allow the controller and/or the bridge device to adjust for such variations encountered in the memory dies. For example, in some embodiments, the controller and/or the bridge device may designate data locations that correspond to physical memory locations that are determined to offer stronger performance to be in SLC mode. Such data locations may be reserved to store critical data such as system data or any other user data that are deemed critical to improve access time and

endurance. As a specific example, some storage systems hard program certain data locations to be in SLC mode (e.g., the first 10 memory blocks) to store system critical data such as system tables. In contrast, in some embodiments, the reserved region offers each controller and/or bridge device in the field the ability to shift the SLC region storing system critical data if the controller and/or bridge device encounters quality issues in the first 10 memory blocks. In some embodiments, the controller and/or bridge device may designate data locations corresponding to higher quality physical memory locations to be in MLC mode to offset the inherent endurance disadvantages of MLC mode vis-à-vis SLC mode. Furthermore, the reserved regions also may serve to give the controller and/or bridge device opportunities to adjust usage of the overall system as the system wears out. In yet another embodiment, the controller and/or bridge device may take advantage of the reserved region to set a range of data locations in SLC mode to store data that can be when the system is in the low knowledge state (e.g., when the system is booting up and not all functionalities are activated yet). These functionalities allow the controller and/or bridge device to dynamically tune the memory regions in response to conditions such as usage patterns, media wear, and external constraints.

#### **IV.C. Variations**

**[0060]** Figures 7A and 7B show example operating mode configuration settings for multiple ranges of memory locations according to various embodiments. Figure 7A shows a general configuration that a reserved region can be between any two regions where the operating modes are set (e.g., a MLC or a SLC mode, or two different MLC modes, or two of the same modes). The configuration shown in Figures 5A and 5B is a particular case of this general configuration. As previously shown in Figure 5A (Illustration 1), the reserved region may initially start at a 0 address. In one embodiment, values indicating the region boundaries are stored and adjusted accordingly to assignment of data locations of the reserved region to accommodate the growth of the adjacent regions. For example, in the scheme of Figure 7A, each MLC/SLC non-reserved region may have a pair of top/bottom values indicating the range of addresses it occupies. Alternatively or in addition, the boundaries of the reserved regions may be kept.

**[0061]** Figure 7B shows the extension of the concept of Figure 7A where there are multiple reserved regions across an address range and the reserved regions are not adjacent to one another. Each pair of top/bottom values tracks the boundaries of the non-reserved regions. Alternatively or in addition, the boundaries of the reserved regions may be kept. The non-reserved regions in Fig. 7B can then be flexibly grown to consume the reserved regions scattered throughout.

**[0062]** The use of reserved region throughout an address range may be useful in many situations. For example, the storage device may be configured to be accessed by multiple buses, with each bus having access to a certain sub-range. In that scenario, each sub-range may be assigned one or more reserved regions so that the flexibility offered by those regions may be provided to the individual controllers associated with the buses. More generally, the various configurations described herein provide any controller the ability to adjust allocation of data locations to MLC/SLC mode to suit its needs.

**[0063]** Figures 8A and 8B illustrate that the different configurations that may be used in different embodiments. Figure 8A shows an embodiment in which all dies (or all planes across the dies) share the same configuration so that changes to the reserved region boundaries take place across the dies. Figure 8B shows an embodiment that allows each die to have a different configuration and be allowed to change individually. In some embodiments, the dies may be divided into subsets where dies in the same subset share the same configuration and are affected in the same way when the boundaries change within the shared configuration.

## **V. Alternative Embodiments; Conclusion**

**[0064]** As used in this application, “non-volatile memory” typically refers to solid-state memory such as NAND flash. However, the systems and methods of this disclosure may also be useful in more conventional hard drives and hybrid drives including both solid-state and hard drive components. As such, while certain internal operations are referred to which typically are associated with solid-state drives, such as “wear leveling” and “garbage collection,” analogous operations for hard drives can also take advantage of some embodiments of this disclosure. Solid-state memory may comprise a wide variety of technologies, such as flash integrated circuits, Chalcogenide RAM (C-RAM), Phase Change Memory (PC-RAM or PRAM),

Programmable Metallization Cell RAM (PMC-RAM or PMCm), Ovonic Unified Memory (OUM), Resistance RAM (RRAM), NAND memory, NOR memory, EEPROM, Ferroelectric Memory (FeRAM), or other discrete NVM (non-volatile memory) chips. The solid-state storage devices (e.g., dies) may be physically divided into planes, blocks, pages, and sectors, as is known in the art. Other forms of storage (e.g., battery backed-up volatile DRAM or SRAM devices, magnetic disk drives, etc.) may additionally or alternatively be used.

**[0065]** While certain embodiments of the inventions 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 methods and systems described herein may be embodied in a variety of other forms. Furthermore, various omissions, substitutions and changes in the form of the methods and systems 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. For example, the various components illustrated in Figures 1D, 2A, 2B, 3, and 4 may be implemented as software and/or firmware on a processor, ASIC/FPGA, or dedicated hardware. For example, those skilled in the art will appreciate that in some embodiments, the actual steps taken in the processes shown in Figure 6 may differ from those shown in the figures. Depending on the embodiment, certain of the steps described in the example above may be removed, others may be added, and the sequence of steps may be altered and/or performed in parallel. Also, the features and attributes of the specific embodiments disclosed above may be combined in different ways to form additional embodiments, all of which fall within the scope of the present disclosure. Although the present disclosure provides certain preferred embodiments and applications, other embodiments that are apparent to those of ordinary skill in the art, including embodiments which do not provide all of the features and advantages set forth herein, are also within the scope of this disclosure. Accordingly, the scope of the present disclosure is intended to be defined only by reference to the appended claims.

**WHAT IS CLAIMED IS:**

1. A non-volatile data storage system, comprising:
  - a controller;
  - an array of one or more solid-state memory storage devices, the memory storage devices including a plurality of multi-level cell (MLC) memory data locations capable of operating in a MLC mode and a single-level cell (SLC) mode; and
  - a bridge device coupled with the array, the bridge device configured to:
    - define a first and second region within at least some of the MLC memory data locations, wherein:
      - the first region includes data locations operating in the SLC or in the MLC mode;
      - the second region includes data locations configured to be assigned to operate in either the SLC or MLC mode;
    - assign data locations in the second region to operate in the SLC or MLC mode in response to a command from the controller; and
    - prevent data access to the data locations in the second region until they are assigned to operate in either the SLC or MLC mode.
2. The data storage system of Claim 1, wherein the second region is defined by an initial starting address and initial ending address.
3. The data storage system of Claim 2, wherein the initial starting address indicates a boundary between the first and the second regions.
4. The data storage system of Claim 2, wherein the bridge device is further configured to assign data locations in the second region to operate in the SLC or MLC mode by at least:
  - adjusting the initial starting address to a new starting address;
  - assigning the data locations located in the second region between the initial starting address and the new starting address to operate in the SLC

mode or in the MLC mode, to match the operating mode of the data locations in the first region that are adjacent to the initial starting address; and enabling data access to the newly assigned data locations.

5. The data storage system of Claim 4, wherein enabling the data access comprises one of the following:

enabling data access to the newly assigned data locations immediately upon receipt of a command from the controller to adjust the initial starting address to the new starting address;

enabling data access to the newly assigned data locations upon receipt of a command from the controller to activate access; and

enabling data access to the newly assigned data locations upon receipt of a command from the controller not specifically for access activation .

6. The data storage system of Claim 4, wherein the bridge device is further configured to:

in response to commands from the controller, successively adjust to a new starting address, wherein each new starting address is greater than a prior starting address; and

after each successive adjustment to the new starting address, assign the data locations located between the new starting address and the prior starting address to operate in a mode to match the operating mode of the data locations in the first region that are adjacent to the prior starting address.

7. The data storage system of Claim 4, wherein the bridge device is further configured to not adjust to the new starting address when it exceeds a threshold set in accordance to a manufacturer warranty constraint.

8. The data storage system of Claim 4, wherein the bridge device is further configured to reset the new starting address to the initial starting address upon a command from the controller.

9. The data storage system of Claim 8, wherein the bridge device is further configured to set a reset flag upon the reset.

10. The data storage system of Claim 2, wherein the bridge device is further configured to define, within the at least some MLC memory data locations, a third region that includes data locations operating in the SLC or in the MLC mode and wherein the initial ending address indicates a boundary between the second and the third regions.

11. The data storage system of Claim 10, wherein the bridge device is further configured to assign data locations in the second region to operate in the SLC or MLC mode by at least:

adjusting the initial ending address to a new ending address;

assigning the data locations located in the second region between the initial ending address and the new ending address to operate in the SLC mode or in the MLC mode, to match the operating mode of the data locations in the third region that are adjacent to the initial ending address; and

enabling data access to the newly assigned data locations.

12. The data storage system of Claim 11, wherein enabling the data access comprises one of the following:

enabling data access to the newly assigned data locations immediately upon receipt of a command from the controller to adjust the initial ending address to the new ending address;

enabling data access to the newly assigned data locations upon receipt of a command from the controller to activate access; and

enabling data access to the newly assigned data locations upon receipt of a command from the controller not specifically for access activation.

13. The data storage system of Claim 11, wherein the bridge device is further configured to:

in response to commands from the controller, successively adjust to a new ending address, wherein each new ending address is smaller than a prior ending address; and

after each successive adjustment to the new ending address, assign the data locations located between the new ending address and the prior ending address to

operate in a mode to match the operating mode of the data locations in the third region that are adjacent to the prior ending address.

14. The data storage system of Claim 11, wherein the bridge device is further configured to not adjust to the new ending address when it exceeds a threshold set in accordance to a manufacturer warranty constraint.

15. The data storage system of Claim 11, wherein the bridge device is further configured to reset the new ending address to the initial ending address upon a command from the controller.

16. The data storage system of Claim 15, wherein the bridge device is further configured to set a reset flag upon the reset.

17. The data storage system of Claim 10, wherein the first region includes data locations operating in the SLC mode and the third region includes data location operating in the MLC mode.

18. The data storage system of Claim 1, wherein the bridge device includes one or more configuration registers and the command from the controller is indicated through setting of the one or more registers.

19. The data storage system of Claim 1, wherein the bridge device is further configured to:

define, within the at least some MLC memory data locations, a fourth region that includes data locations configured to be assigned to operate in either the SLC or MLC mode;

assign data locations in the fourth region to operate in the SLC or MLC mode in response to a command from the controller; and

prevent data access to the data locations in the fourth region until they are assigned to operate in either the SLC or MLC mode.

20. The data storage system of Claim 19, wherein the second and the fourth regions are not adjacent to each other.



21. The data storage system of Claim 1 further comprising rotating magnetic media under control of the controller.

22. A method of operating a bridge device in a non-volatile data storage system comprising (1) a controller, (2) an array of one or more solid-state memory storage devices, the memory storage devices including a plurality of multi-level cell (MLC) memory data locations capable of operating in a MLC mode and a single-level cell (SLC) mode, and (3) the bridge device coupled with the array, the method comprising:

defining a first and second region within at least some of the MLC memory data locations, wherein:

the first region includes data locations operating in the SLC or in the MLC mode;

the second region includes data locations configured to be assigned to operate in either the SLC or MLC mode;

assigning data locations in the second region to operate in the SLC or MLC mode in response to a command from the controller; and

preventing data access to the data locations in the second region until they are assigned to operate in either the SLC or MLC mode.

23. The method of Claim 22, wherein the second region is defined by an initial starting address and initial ending address.

24. The method of Claim 23, wherein the initial starting address indicates a boundary between the first and the second regions.

25. The method of Claim 23, wherein assigning data locations in the second region to operate in the SLC or MLC mode further comprises:

adjusting the initial starting address to a new starting address;

assigning the data locations located in the second region between the initial starting address and the new starting address to operate in the SLC mode or in the MLC mode, to match the operating mode of the data locations in the first region that are adjacent to the initial starting address; and

enabling data access to the newly assigned data locations.

26. The method of Claim 25, wherein enabling the data access comprises one of the following:

enabling data access to the newly assigned data locations immediately upon receipt of a command from the controller to adjust the initial starting address to the new starting address;

enabling data access to the newly assigned data locations upon receipt of a command from the controller to activate access; and

enabling data access to the newly assigned data locations upon receipt of a command from the controller not specifically for access activation.

27. The method of Claim 25, further comprising:

in response to commands from the controller, successively adjusting to a new starting address, wherein each new starting address is greater than a prior starting address; and

after each successive adjustment to the new starting address, assigning the data locations located between the new starting address and the prior starting address to operate in a mode to match the operating mode of the data locations in the first region that are adjacent to the prior starting address.

28. The method of Claim 25, further comprising not adjusting to the new starting address when it exceeds a threshold set in accordance to a manufacturer warranty constraint.

29. The method of Claim 25, further comprising resetting the new starting address to the initial starting address upon a command from the controller.

30. The method of Claim 29, further comprising setting a reset flag upon the resetting.

31. The method of Claim 23, further comprising defining, within the at least some MLC memory data locations, a third region that includes data locations operating in the SLC or in the MLC mode and wherein the initial ending address indicates a boundary between the second and the third regions.

32. The method of Claim 31, wherein assigning data locations in the second region to operate in the SLC or MLC mode comprises:

- adjusting the initial ending address to a new ending address;
- assigning the data locations located in the second region between the initial ending address and the new ending address to operate in the SLC mode or in the MLC mode, to match the operating mode of the data locations in the third region that are adjacent to the initial ending address; and
- enabling data access to the newly assigned data locations.

33. The method of Claim 32, wherein enabling the data access comprises one of the following:

- enabling data access to the newly assigned data locations immediately upon receipt of a command from the controller to adjust the initial ending address to the new ending address;
- enabling data access to the newly assigned data locations upon receipt of a command from the controller to activate access; and
- enabling data access to the newly assigned data locations upon receipt of a command from the controller not specifically for access activation.

34. The method of Claim 32, further comprising:

- in response to commands from the controller, successively adjusting to a new ending address, wherein each new ending address is smaller than a prior ending address; and
- after each successive adjustment to the new ending address, assigning the data locations located between the new ending address and the prior ending address to operate in a mode to match the operating mode of the data locations in the third region that are adjacent to the prior ending address.

35. The method of Claim 32, further comprising not adjusting to the new ending address when it exceeds a threshold set in accordance to a manufacturer warranty constraint.

36. The method of Claim 32, further comprising resetting the new ending address to the initial ending address upon a command from the controller.

37. The method of Claim 36, further comprising setting a reset flag upon the resetting.

38. The method of Claim 31, wherein the first region includes data locations operating in the SLC mode and the third region includes data location operating in the MLC mode.

39. The method of Claim 22, wherein the bridge device includes one or more configuration registers and the command from the controller is indicated through setting of the one or more registers.

40. The method of Claim 22, further comprising:  
defining, within the at least some MLC memory data locations, a fourth region that includes data locations configured to be assigned to operate in either the SLC or MLC mode;

assigning data locations in the fourth region to operate in the SLC or MLC mode in response to a command from the controller; and

preventing data access to the data locations in the fourth region until they are assigned to operate in either the SLC or MLC mode.

41. The method of Claim 40, wherein the second and the fourth regions are not adjacent to each other.

42. The method of Claim 22, wherein the data storage system further comprises rotating magnetic media under control of the controller.

43. A bridge device configured to be coupled with (1) an array of one or more solid-state memory storage devices, the memory storage devices including a plurality of multi-level cell (MLC) memory data locations capable of operating in a MLC mode and a single-level cell (SLC) mode and (2) a controller in a non-volatile data storage system, the bridge device configured to:

define a first and second region within at least some of the MLC memory data locations, wherein:

the first region includes data locations operating in the SLC or in the MLC mode;

the second region includes data locations configured to be assigned to operate in either the SLC or MLC mode;

assign data locations in the second region to operate in the SLC or MLC mode in response to a command from the controller; and

prevent data access to the data locations in the second region until they are assigned to operate in either the SLC or MLC mode.

44. A non-volatile data storage system, comprising:

a controller;

an array of one or more solid-state memory storage devices, the memory storage devices including a plurality of multi-level cell (MLC) memory data locations capable of operating in a plurality of MLC modes and a single-level cell (SLC) mode; and

a bridge device coupled with the array, the bridge device configured to:

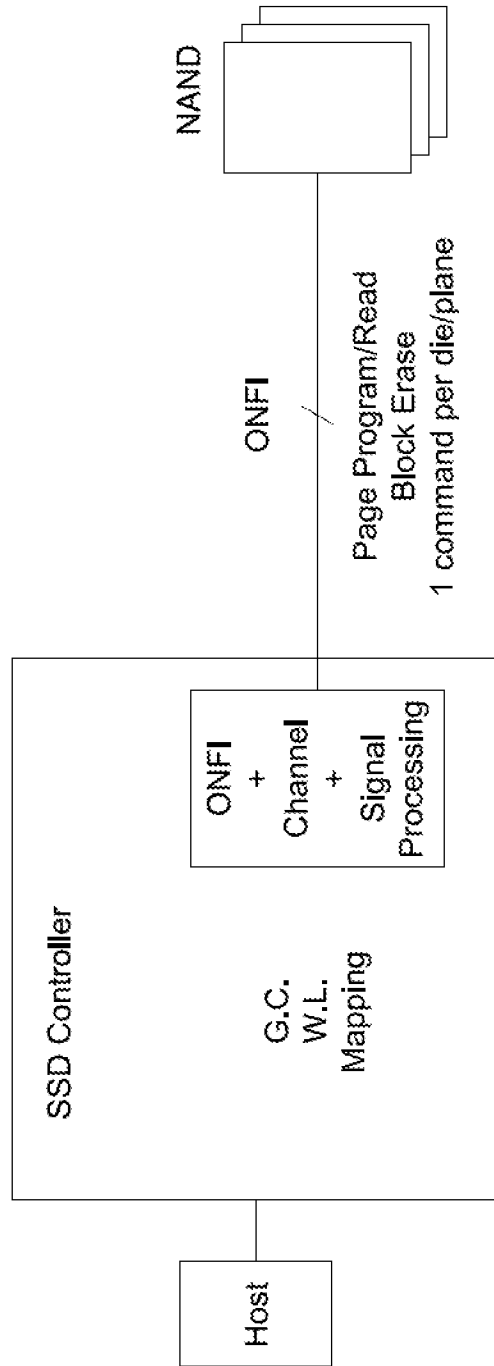
define a first and second region within at least some of the MLC memory data locations, wherein:

the first region includes data locations operating in the SLC mode or in one of the plurality of MLC modes;

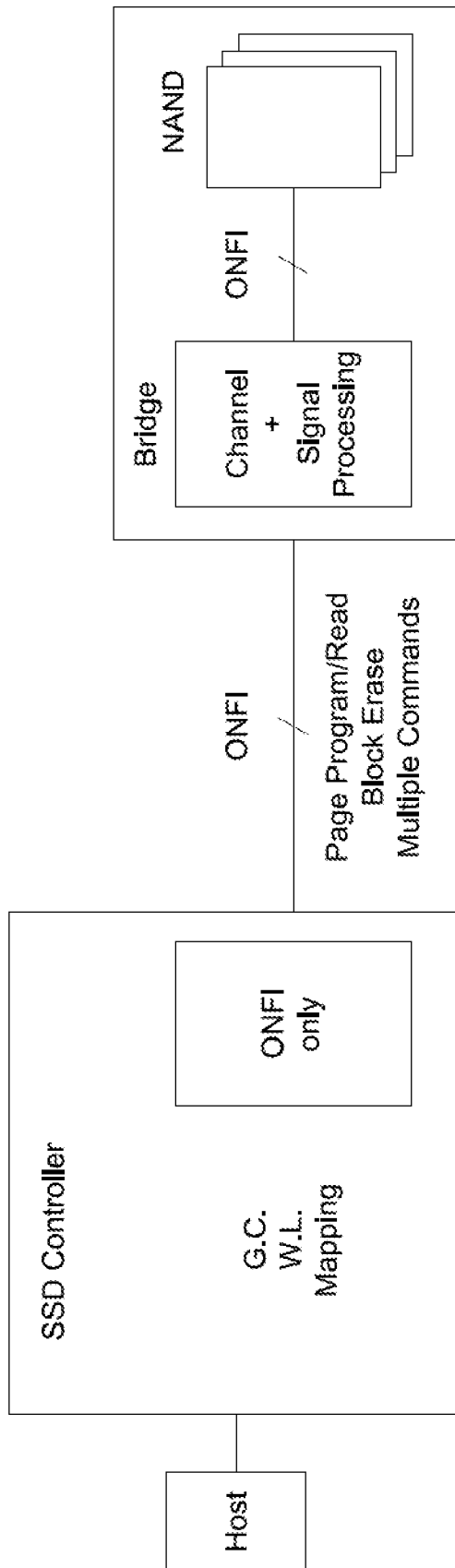
the second region includes data locations configured to be assigned to operate in the SLC mode or one of the plurality of MLC modes;

assign data locations in the second region to operate in the SLC or one of the plurality MLC modes in response to a command from the controller; and

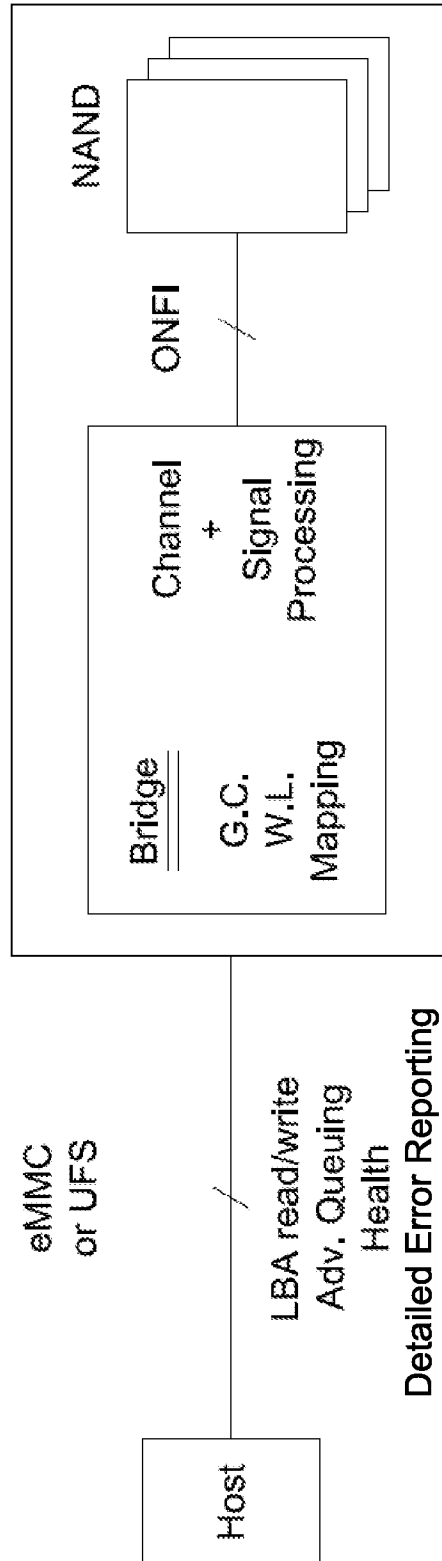
prevent data access to the data locations in the second region until they are assigned to operate in either the SLC or one of the plurality of MLC modes.



**FIGURE 1A**

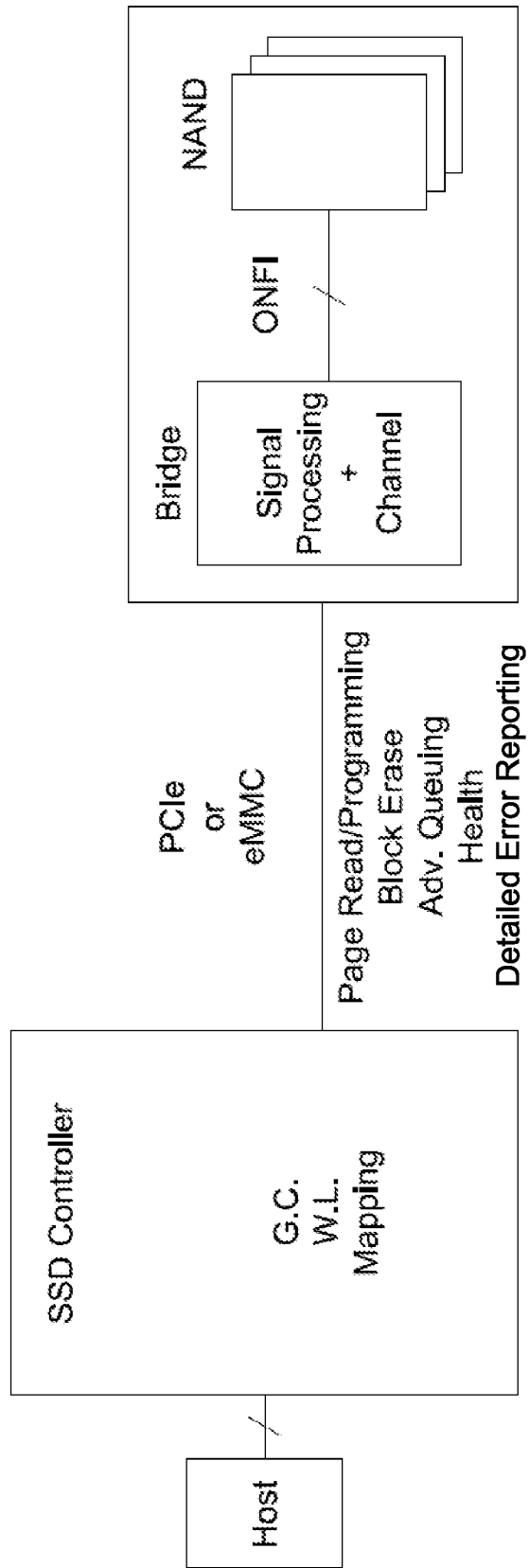


**FIGURE 1B**



**FIGURE 1C**





**FIGURE 1D**

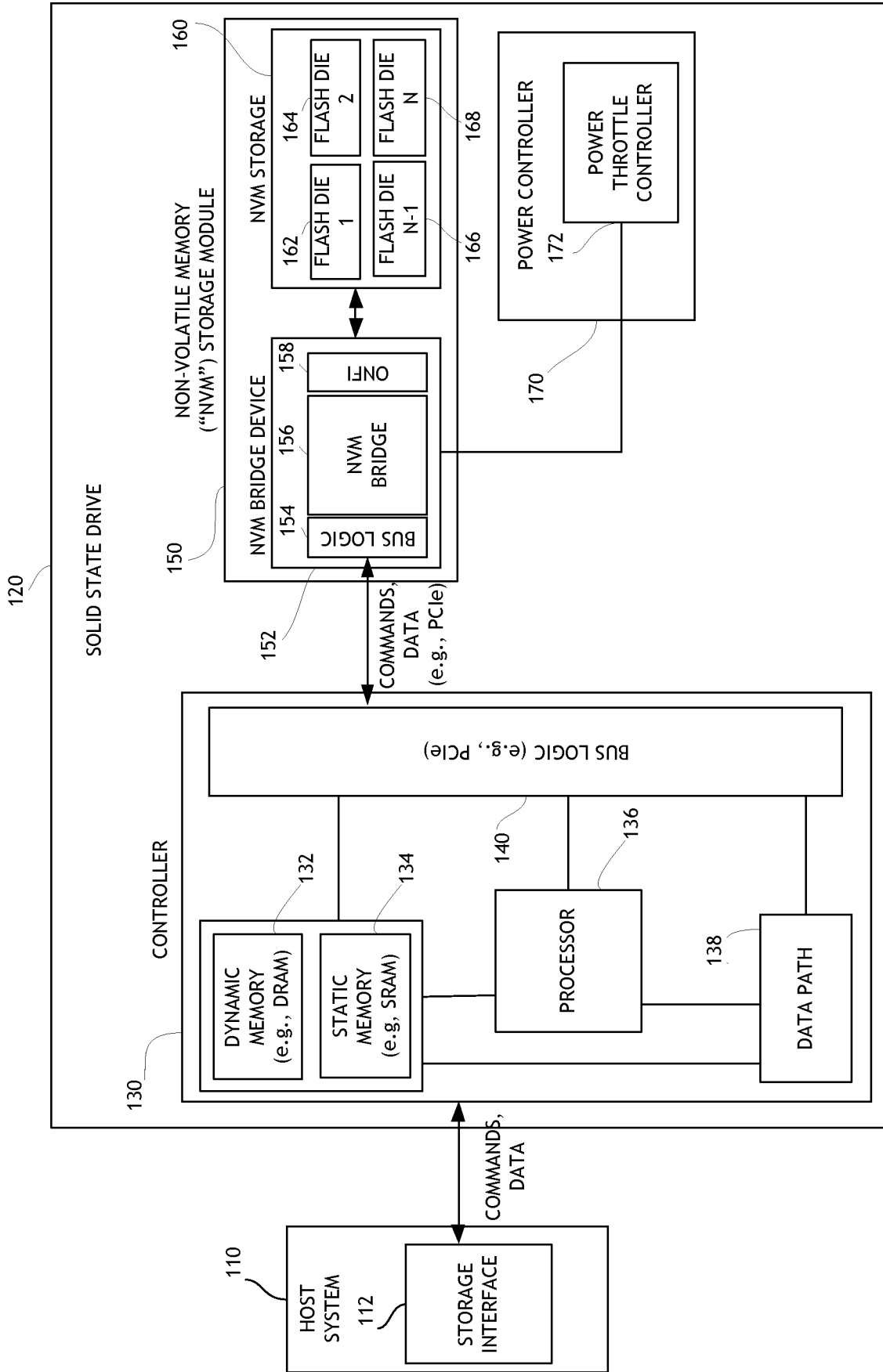
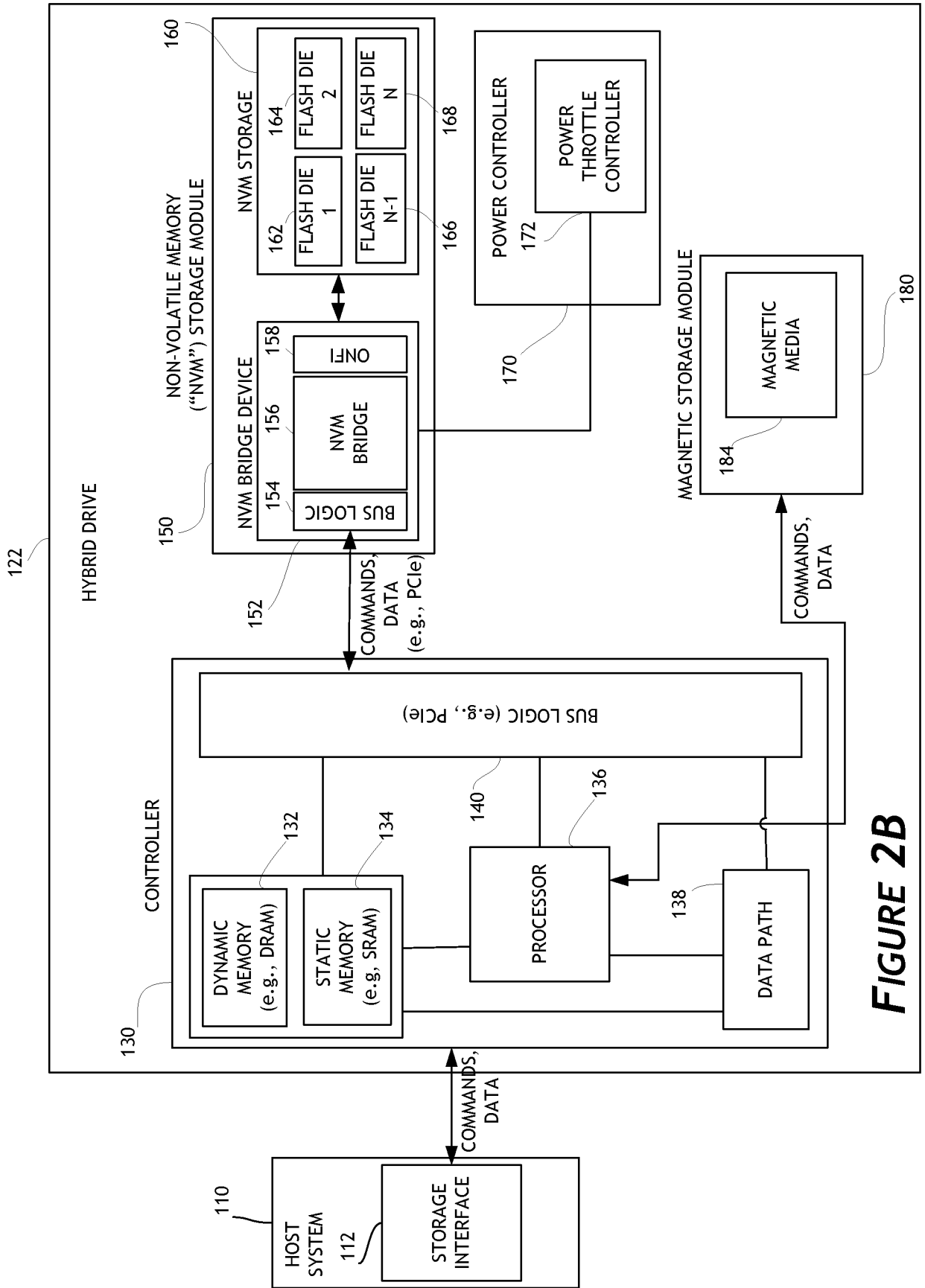


FIGURE 2A



**FIGURE 2B**

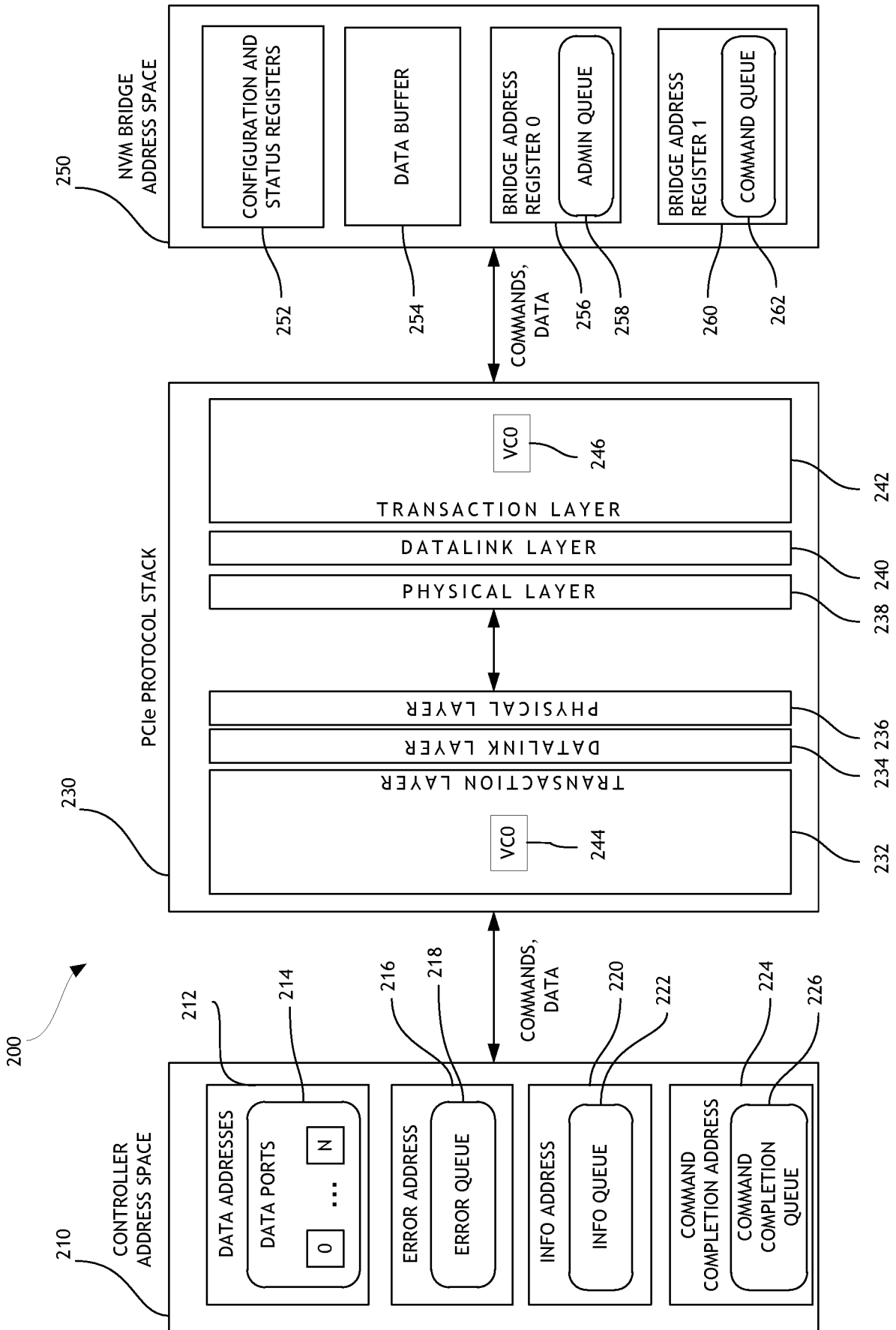
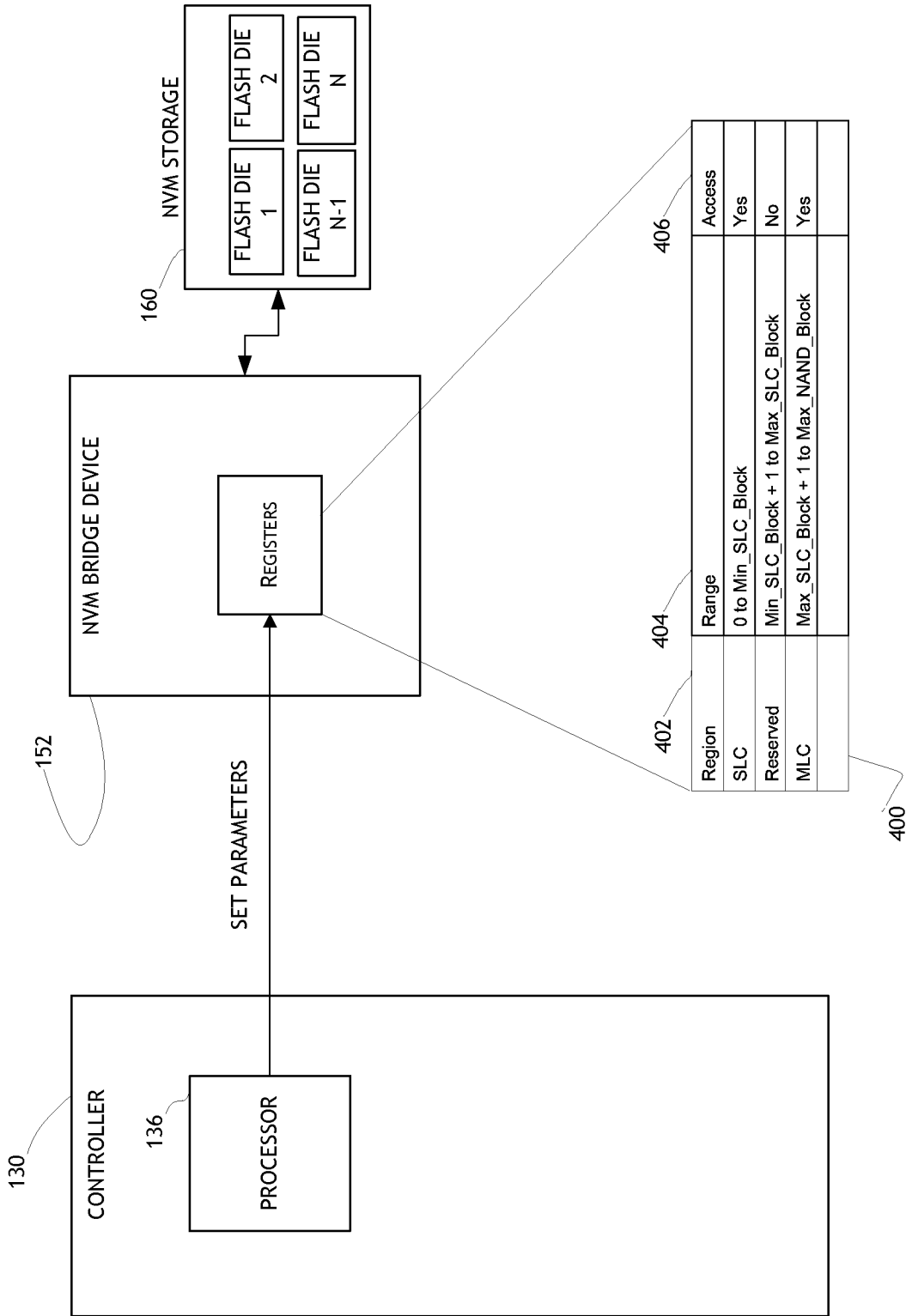
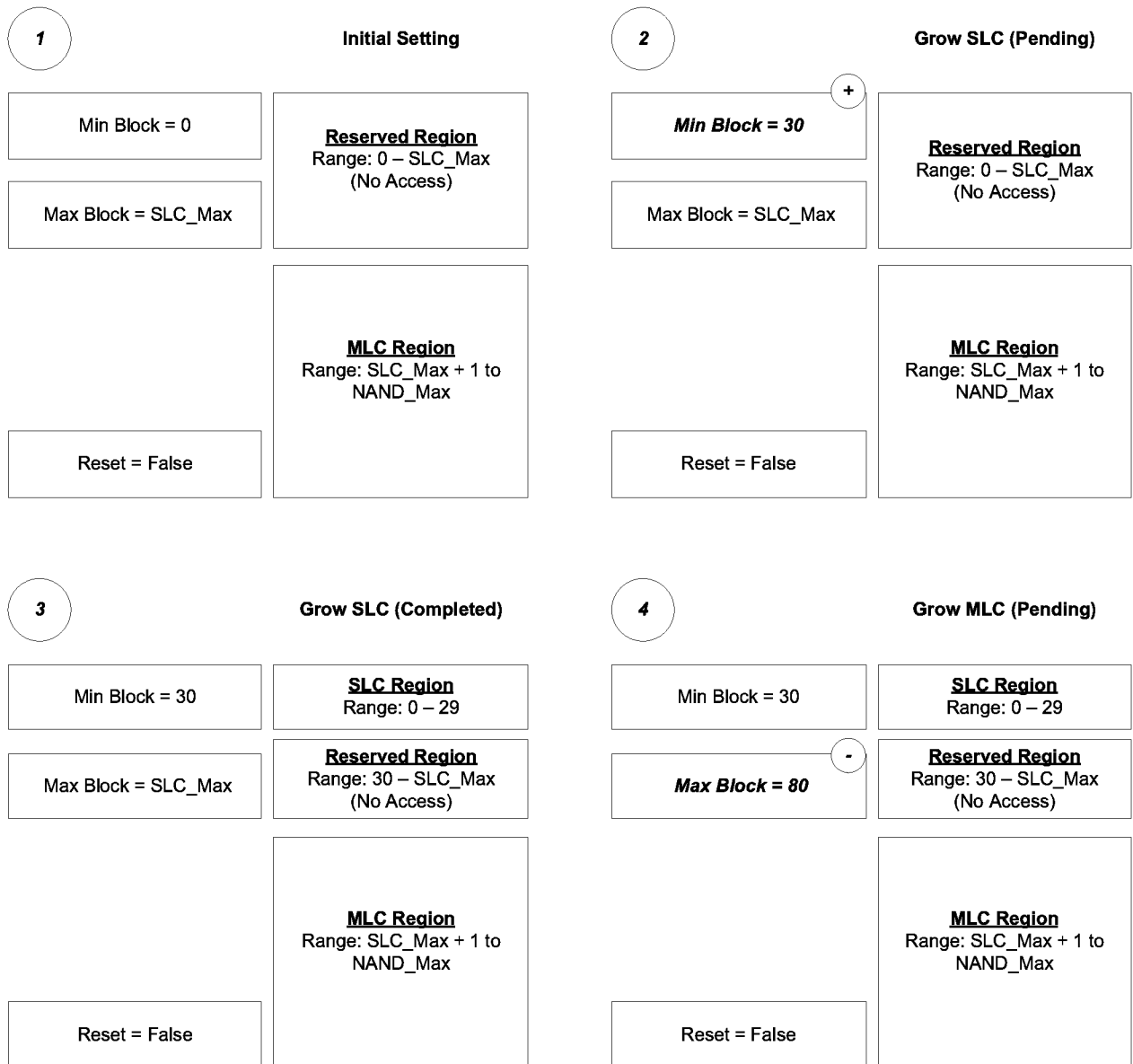


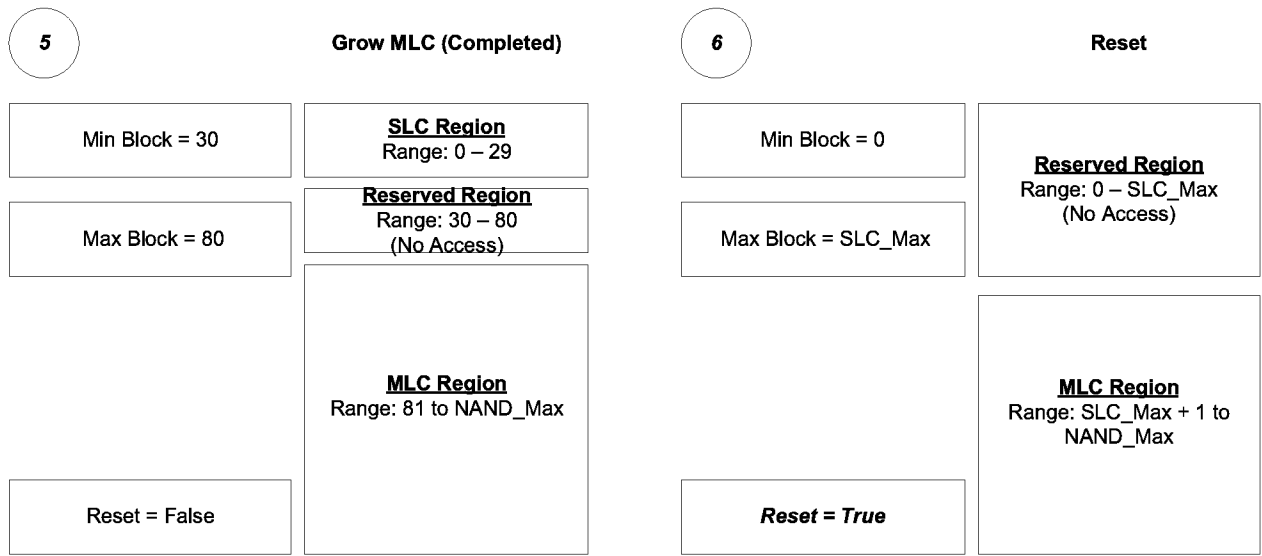
FIGURE 3



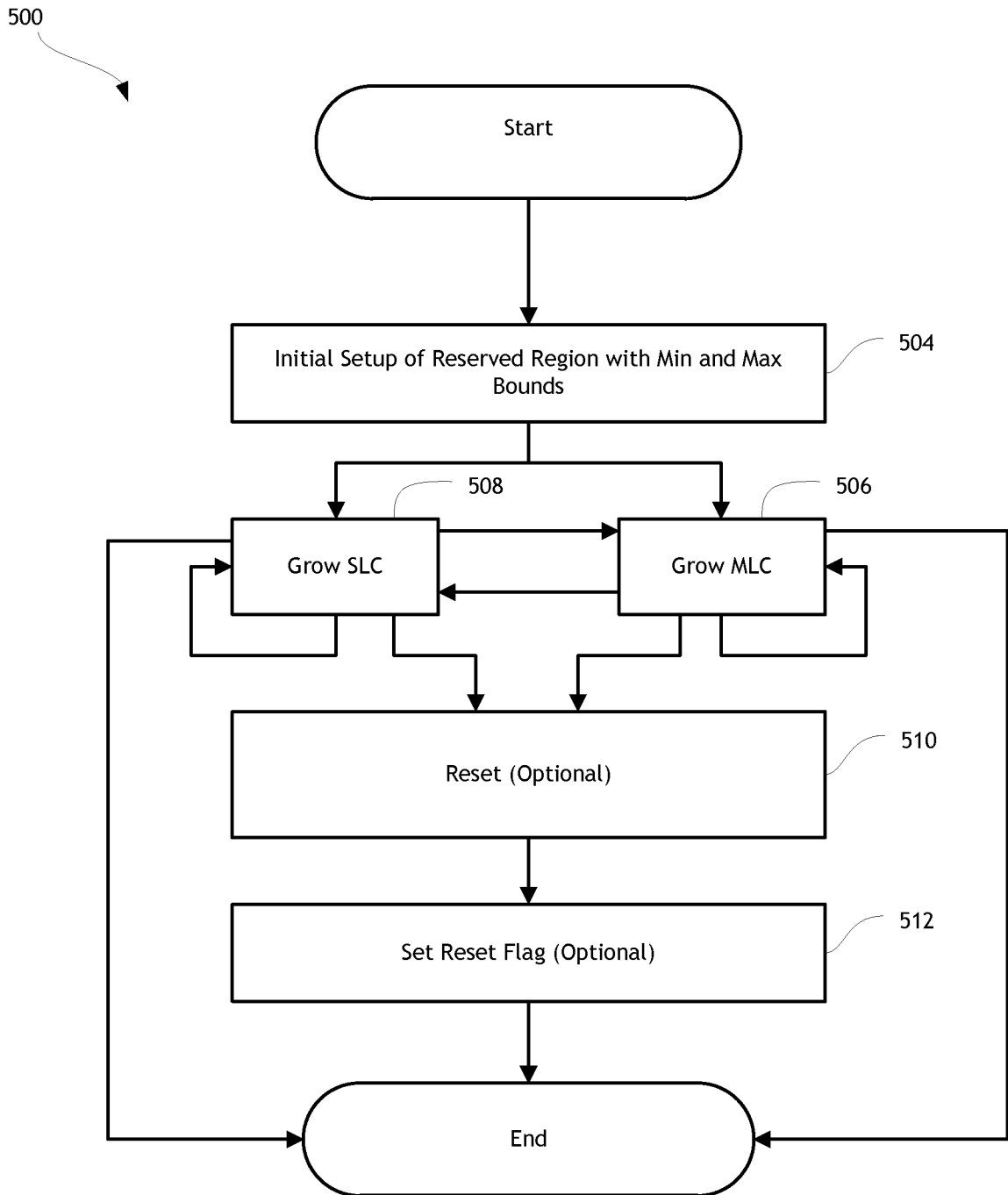
**FIGURE 4**



**FIGURE 5A**

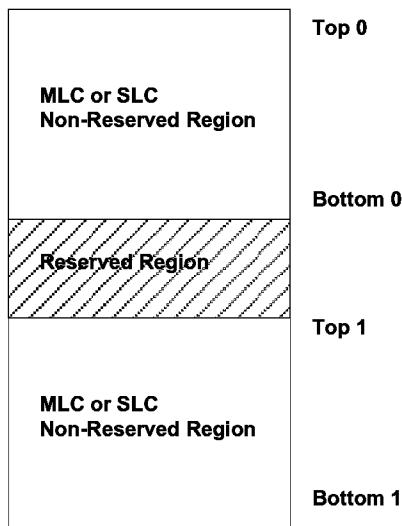


**FIGURE 5B**

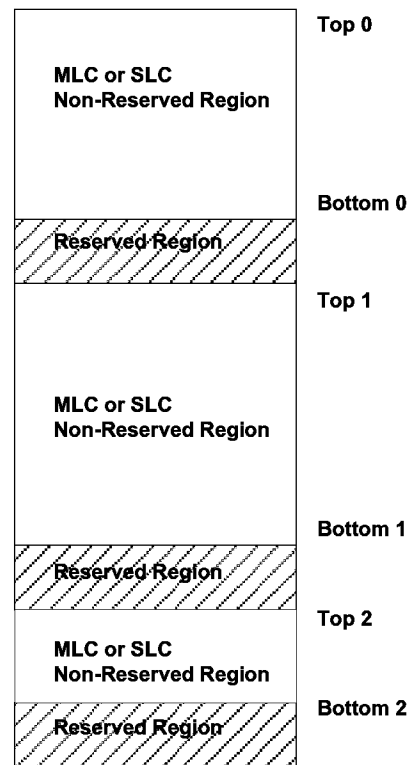


**FIGURE 6**

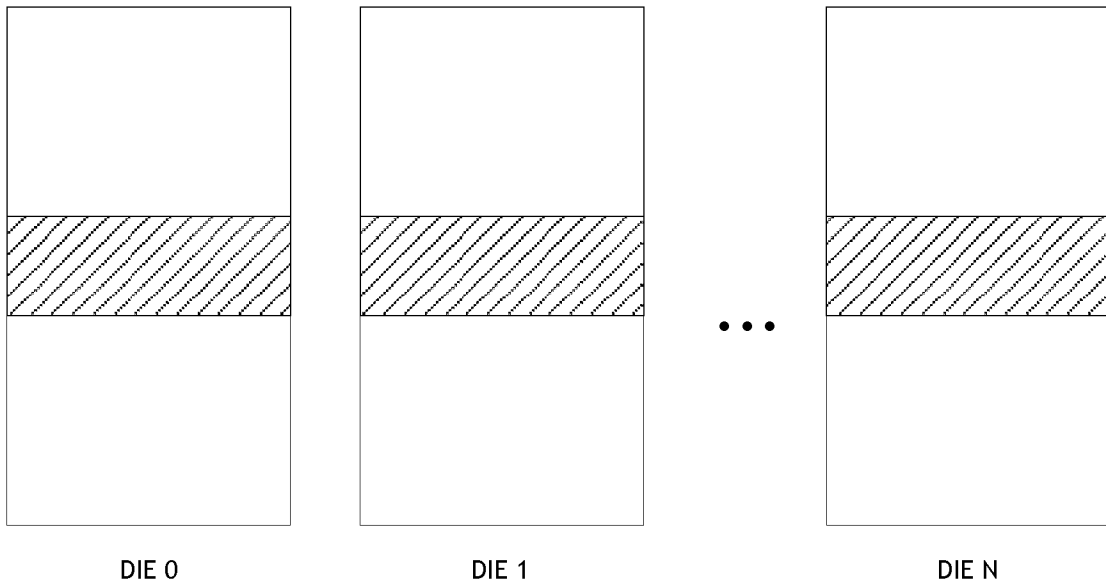




**FIGURE 7A**

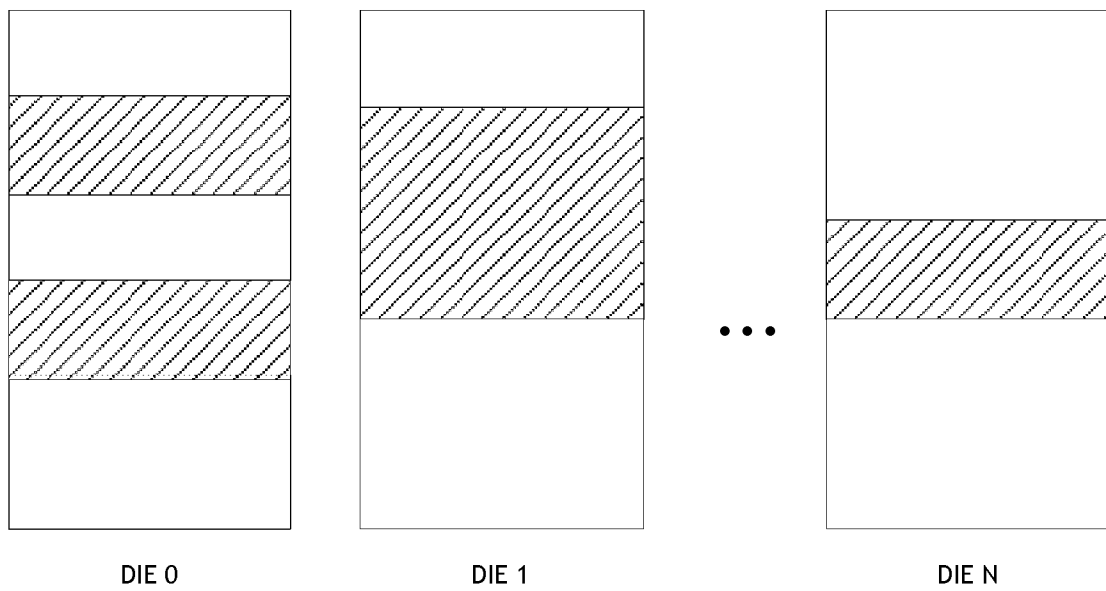


**FIGURE 7B**



**FIGURE 8A**

 Reserved Region  
 Non-Reserved Region – SLC or MLC



**FIGURE 8B**

**A. CLASSIFICATION OF SUBJECT MATTER****G06F 12/00(2006.01)i, G06F 13/14(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 12/00; G06F 12/02; H03M 13/05; G06F 11/10; G11C 29/00; G06F 13/14

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) &amp; Keywords: non-volatile, solid-state, multi-level cell, single-level cell, reserved region, location, data access, prevent, assign, bridge device, and similar terms.

**C. DOCUMENTS CONSIDERED TO BE RELEVANT**

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 2010-0122016 A1 (GIULIO MAROTTA et al.) 13 May 2010 See paragraphs [0018]-[0033]; claims 1-8; and figures 1-4.	1-44
A	US 2012-0240012 A1 (ANTHONY D. WEATHERS et al.) 20 September 2012 See paragraphs [0006]-[0008], [0015]-[0023], and [0032]-[0042]; claim 8; and figures 1 and 4.	1-44
A	US 2008-0215800 A1 (CHARLES C. LEE et al.) 04 September 2008 See paragraphs [0026]-[0030] and [0034]-[0037]; and figures 2 and 4.	1-44
A	US 2008-0126680 A1 (YANG-SUP LEE et al.) 29 May 2008 See paragraphs [0024]-[0036] and figures 1-4.	1-44
A	US 8266503 B2 (ROBERT WOOD et al.) 11 September 2012 See column 5, line 1 - column 8, line 19; and figures 1-3.	1-44

 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

"&amp;" document member of the same patent family

Date of the actual completion of the international search

29 April 2014 (29.04.2014)

Date of mailing of the international search report

**29 April 2014 (29.04.2014)**

Name and mailing address of the ISA/KR

International Application Division  
Korean Intellectual Property Office  
189 Cheongsu-ro, Seo-gu, Daejeon Metropolitan City, 302-701,  
Republic of Korea

Facsimile No. +82-42-472-7140

Authorized officer

NHO, Ji Myong

Telephone No. +82-42-481-8528



**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No.

**PCT/US2014/011352**

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2010-0122016 A1	13/05/2010	US 2013-227203 A1 US 8407400 B2 US 8667215 B2	29/08/2013 26/03/2013 04/03/2014
US 2012-0240012 A1	20/09/2012	US 8656256 B2	18/02/2014
US 2008-0215800 A1	04/09/2008	AU 1999-29734 A1 AU 1999-60212 A1 AU 1999-61310 A1 AU 2000-20200 A1 AU 2001-26352 A1 AU 2001-269765 A8 AU 2001-47725 A1 AU 2001-69765 A1 AU 2002-15644 A1 AU 2002-15644 B2 AU 2002-337822 A8 AU 2002-362087 A1 AU 2002-362087 A8 AU 2003-267265 A1 AU 2003-301638 A1 AU 2003-301638 A8 CA 2280865 A1 CA 2280865 C CA 2345300 A1 CA 2345300 C CA 2345463 A1 CA 2345463 C CA 2373569 A1 CA 2402552 A1 CA 2402552 C CA 2412426 A1 CA 2412426 C CA 2461338 A1 CA 2461338 C CA 2469001 A1 CA 2469001 C CA 2499993 A1 CA 2499993 C CA 2521079 A1 CA 2762083 A1 CA 2762083 C CN 100472856 C CN 100487678 C CN 100501702 C CN 101097551 A CN 101118783 A CN 101122865 A CN 101122865 B	06/09/1999 21/03/2000 21/03/2000 03/04/2000 16/07/2001 17/12/2001 03/10/2001 17/12/2001 29/08/2002 06/07/2006 22/04/2003 23/06/2003 23/06/2003 08/04/2004 13/05/2004 13/05/2004 27/08/1998 12/08/2008 09/03/2000 24/01/2012 23/03/2000 18/11/2008 28/08/2002 27/09/2001 02/10/2007 13/12/2001 04/09/2007 17/04/2003 20/12/2011 19/06/2003 17/06/2008 01/04/2004 06/08/2013 04/11/2004 09/03/2000 06/08/2013 25/03/2009 13/05/2009 17/06/2009 02/01/2008 06/02/2008 13/02/2008 14/07/2010

**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No.

**PCT/US2014/011352**

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		CN 101122887 A	13/02/2008
		CN 101398764 A	01/04/2009
		CN 101398785 A	01/04/2009
		CN 101399075 A	01/04/2009
		CN 101399076 A	01/04/2009
		CN 101409111 A	15/04/2009
		CN 101425324 A	06/05/2009
		CN 101630375 A	20/01/2010
		CN 101644995 A	10/02/2010
		CN 101644995 B	28/12/2011
		CN 101727976 A	09/06/2010
		CN 101727976 B	19/09/2012
		CN 101923512 A	22/12/2010
		CN 101923512 B	20/03/2013
		CN 102469715 A	23/05/2012
		CN 1325584 C	11/07/2007
		CN 1373166 A	09/10/2002
		CN 1806356 A	19/07/2006
		EP 0937764 A1	25/08/1999
		EP 0985007 A1	15/03/2000
		EP 0985007 B1	17/05/2006
		EP 0985007 B2	03/11/2010
		EP 1007308 A1	14/06/2000
		EP 1007308 B1	12/11/2003
		EP 1127098 A1	29/08/2001
		EP 1236776 A1	04/09/2002
		EP 1236776 B1	21/12/2005
		EP 1236776 B2	25/11/2009
		EP 1268054 A1	02/01/2003
		EP 1268054 B1	05/12/2012
		EP 1309396 A2	14/05/2003
		EP 1309396 B1	21/09/2011
		EP 1358124 A2	05/11/2003
		EP 1358124 B1	11/03/2009
		EP 1386708 A2	04/02/2004
		EP 1386708 A3	04/06/2008
		EP 1448725 A2	25/08/2004
		EP 1450725 A2	01/09/2004
		EP 1542592 A1	22/06/2005
		EP 1559165 A2	03/08/2005
		EP 1629549 A2	01/03/2006
		EP 1629549 B1	26/10/2011
		EP 1889970 A1	20/02/2008
		JP 2001-005945 A	12/01/2001
		JP 2001-118046 A	27/04/2001
		JP 2001-513697 A	04/09/2001
		JP 2001-513828 A	04/09/2001
		JP 2002-309151 A	23/10/2002
		JP 2002-523610 A	30/07/2002
		JP 2003-527735 A	16/09/2003

**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No.

**PCT/US2014/011352**

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		JP 2003-528419 A	24/09/2003
		JP 2004-507341 A	11/03/2004
		JP 2005-511186 A	28/04/2005
		JP 2005-537386 A	08/12/2005
		JP 2006-508501 A	09/03/2006
		JP 2006-524898 A	02/11/2006
		JP 2007-332523 A	27/12/2007
		JP 2008-173635 A	31/07/2008
		JP 2009-084577 A	23/04/2009
		JP 2010-272544 A	02/12/2010
		JP 2011-335658 A	07/12/1999
		JP 3274433 B2	15/04/2002
		JP 3338417 B2	28/10/2002
		JP 4097944 B2	11/06/2008
		JP 4178170 B2	12/11/2008
		JP 4471342 B2	02/06/2010
		JP 4704563 B2	15/06/2011
		KR 10-0797125 B1	22/01/2008
		KR 10-0893564 B1	17/04/2009
		KR 10-0988295 B1	18/10/2010
		TW 200813713 A	16/03/2008
		TW 200842887 A	01/11/2008
		TW 200915080 A	01/04/2009
		TW 200917017 A	16/04/2009
		TW 200926394 A	16/06/2009
		TW 201005652 A	01/02/2010
		TW 1351599 B	01/11/2011
		TW 1351605 B	01/11/2011
		US 06103393 A	15/08/2000
		US 06153123 A	28/11/2000
		US 06159267 A	12/12/2000
		US 6168731 B1	02/01/2001
		US 6180029 B1	30/01/2001
		US 6193908 B1	27/02/2001
		US 6197218 B1	06/03/2001
		US 6210604 B1	03/04/2001
		US 6277169 B1	21/08/2001
		US 6316100 B1	13/11/2001
		US 6338809 B1	15/01/2002
		US 6360562 B1	26/03/2002
		US 6507349 B1	14/01/2003
		US 6547130 B1	15/04/2003
		US 6555022 B2	29/04/2003
		US 6592608 B2	15/07/2003
		US 6602439 B1	05/08/2003
		US 6623856 B1	23/09/2003
		US 6627115 B2	30/09/2003
		US 6635348 B1	21/10/2003
		US 6645398 B1	11/11/2003
		US 6660185 B2	09/12/2003

**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No.

**PCT/US2014/011352**

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		US 6660680 B1	09/12/2003
		US 6679937 B1	20/01/2004
		US 6685727 B2	03/02/2004
		US 6689186 B1	10/02/2004
		US 6699304 B1	02/03/2004
		US 6709408 B2	23/03/2004
		US 6730245 B2	04/05/2004
		US 6753108 B1	22/06/2004
		US 6770226 B2	03/08/2004
		US 6780350 B1	24/08/2004
		US 6854984 B1	15/02/2005
		US 6866929 B2	15/03/2005
		US 6872185 B2	29/03/2005
		US 6874044 B1	29/03/2005
		US 6875372 B1	05/04/2005
		US 6887928 B2	03/05/2005
		US 6890309 B2	10/05/2005
		US 6899329 B2	31/05/2005
		US 6908440 B2	21/06/2005
		US 6911412 B2	28/06/2005
		US 6955601 B2	18/10/2005
		US 6967183 B2	22/11/2005
		US 6991754 B2	31/01/2006
		US 6993934 B2	07/02/2006
		US 6994802 B2	07/02/2006
		US 7001410 B2	21/02/2006
		US 7004794 B2	28/02/2006
		US 7004994 B2	28/02/2006
		US 7005085 B2	28/02/2006
		US 7021971 B2	04/04/2006
		US 7022261 B2	04/04/2006
		US 7035110 B1	25/04/2006
		US 7037451 B2	02/05/2006
		US 7044802 B2	16/05/2006
		US 7066976 B2	27/06/2006
		US 7067070 B2	27/06/2006
		US 7069369 B2	27/06/2006
		US 7073010 B2	04/07/2006
		US 7082056 B2	25/07/2006
		US 7083747 B2	01/08/2006
		US 7087198 B2	08/08/2006
		US 7087341 B2	08/08/2006
		US 7094074 B2	22/08/2006
		US 7094370 B2	22/08/2006
		US 7095617 B1	22/08/2006
		US 7097686 B2	29/08/2006
		US 7098163 B2	29/08/2006
		US 7103684 B2	05/09/2006
		US 7103765 B2	05/09/2006
		US 7104848 B1	12/09/2006

**INTERNATIONAL SEARCH REPORT**

Information on patent family members

International application No.

**PCT/US2014/011352**

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
		US 7108560 B1	19/09/2006
		US 7125287 B1	24/10/2006
		US 7125919 B2	24/10/2006
		US 7128852 B2	31/10/2006
		US 7130958 B2	31/10/2006
		US 7138159 B2	21/11/2006
		US 7138354 B2	21/11/2006
		US 7141528 B2	28/11/2006
		US 7150920 B2	19/12/2006
		US 7172663 B2	06/02/2007
		US 7174628 B1	13/02/2007
		US 7182646 B1	27/02/2007
		US 7186147 B1	06/03/2007
		US 7211345 B2	01/05/2007
		US 7226547 B2	05/06/2007
		US 7229572 B2	12/06/2007
		US 7243185 B2	10/07/2007
		US 7249978 B1	31/07/2007
		US 7255954 B2	14/08/2007
		US 7257714 B1	14/08/2007
		US 7259967 B2	21/08/2007
		US 7264992 B2	04/09/2007
		US 7269004 B1	11/09/2007
		US 7296345 B1	20/11/2007
		US 7297024 B2	20/11/2007
		US 7299316 B2	20/11/2007
		US 7301776 B1	27/11/2007
		US 7316725 B2	08/01/2008
		US 7316790 B2	08/01/2008
		US 7318117 B2	08/01/2008
		US 7333364 B2	19/02/2008
US 2008-0126680 A1	29/05/2008	KR 10-0833188 B1	28/05/2008
		KR 10-2008-0040426 A	08/05/2008
		US 2010-0153632 A1	17/06/2010
		US 7676626 B2	09/03/2010
		US 8214582 B2	03/07/2012
US 8266503 B2	11/09/2012	US 2012-173827 A1	05/07/2012
		US 2013-003457 A1	03/01/2013
		US 8527841 B2	03/09/2013