(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
1 May 2014 (01.05.2014)

WIPO | PCT

(10) International Publication Number
WO 2014/065967 A1

(51) International Patent Classification:
G11C 29/42 (2006.01) *G11C 16/34* (2006.01)

(21) International Application Number:
PCT/US2013/061249

(22) International Filing Date:
23 September 2013 (23.09.2013)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
13/659,099 24 October 2012 (24.10.2012) US

(71) Applicant: **WESTERN DIGITAL TECHNOLOGIES, INC.** [US/US]; 3355 Michelson Drive, Suite 100, Irvine, CA 92612 (US).

(72) Inventor: **LU, Guangming**; c/o Western Digital Technologies, Inc., 3355 Michelson Drive, Suite 100, Irvine, CA 92612 (US).

(74) Agent: **DELANEY, Karoline A.**; Knobbe Martens Olson & Bear, LLP, 2040 Main Street, 14th Floor, Irvine, CA 92614 (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, 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**: ADAPTIVE ERROR CORRECTION CODES FOR DATA STORAGE SYSTEMS

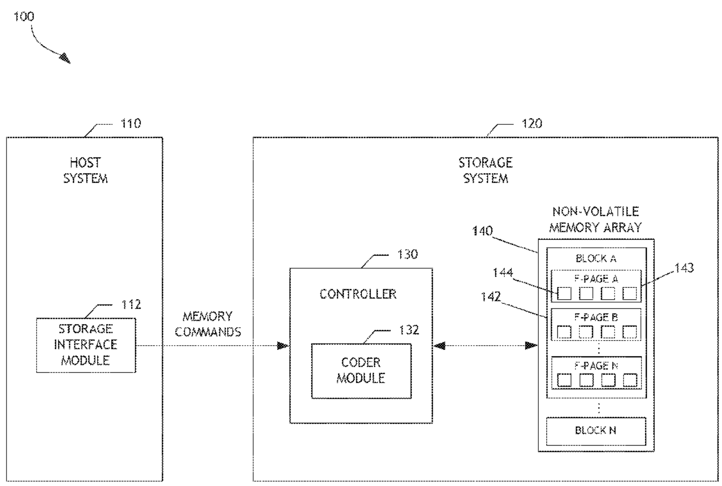

FIGURE 1

(57) **Abstract**: A data storage system configured to adaptively code data is disclosed, in one embodiment, a data storage system controller determines a common memory page size, such as an E-page size, for a non-volatile memory array. Based on the common memory page size, the controller selects a low-density parity-check (LDPC) code word length from a plurality of pre-defined LDPC code word lengths. The controller determines LDPC coding parameters for coding data written to or read from the memory array based on the selected LDPC code word length. By using the plurality of pre-defined LDPC code word lengths, the data storage system can support multiple non-volatile memory page formats, including memory page formats in which the common memory page size does not equal any LDPC code word length of the plurality of pre-defined LDPC code word lengths. Flexibility and efficiency of data coding can thereby be achieved.

WO 2014/065967 A1

ADAPTIVE ERROR CORRECTION CODES FOR DATA STORAGE SYSTEMS

BACKGROUND

Technical Field

[0001] This disclosure relates to data storage systems, such as solid state drives, for computer systems. More particularly, the disclosure relates to adaptive error correction codes for data storage systems.

Description of the Related Art

[0002] Non-volatile memory arrays often have limited endurance. The endurance of the memory array is typically contingent on usage pattern and wear. In addition, endurances depend on a particular type of the non-volatile memory array. For example, memory arrays with multi-level cell (MLC) NAND media typically have a lower endurance than memory arrays with single-level cell (SLC) NAND media. To protect user data stored to memory arrays from corruption, which may be caused by a diminished endurance, parity data can be determined and stored along with user data to facilitate error detection and/or correction.

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] Figure 1 illustrates a storage system that adaptively codes data according to one embodiment of the invention.

[0005] Figure 2 is a flow diagram illustrating a process of determining coding parameters for coding data according to one embodiment of the invention.

[0006] Figure 3 is a table illustrating relationships between low-density parity-check (LDPC) coding parameters and LDPC code word lengths according to one embodiment of the invention.

[0007] Figures 4A-4C are diagrams illustrating padded user data and parity data according to one embodiment of the invention.

[0008] Figure 5 is a flow diagram illustrating a process of adaptive code shortening according to one embodiment of the invention.

[0009] Figure 6 is a flow diagram illustrating a process of adjusting a code rate according to one embodiment of the invention.

DETAILED DESCRIPTION OF SPECIFIC EMBODIMENTS

[0010] While certain embodiments are described, these embodiments are presented by way of example only, and are not intended to limit the scope of protection. 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 scope of protection.

[0011] In some embodiments, "coding" or "to code" data as used in this disclosure refer to the process of encoding data and/or the process of decoding data.

Overview

[0012] Storage systems such as solid state drives typically include one or more controllers coupled with non-volatile memory arrays. It is commonplace for such controllers to be designed/manufactured by one party and the non-volatile memory arrays to be designed/manufactured by another party. In addition, memory arrays from different manufacturers tend to have different internal formatting, including different memory page formats, where each memory page format corresponds to one of multiple memory pages sizes. Also, for costs and various other competitive reasons, storage system manufacturers typically use memory arrays from different manufacturers. For example, a storage system manufacturer may use one brand of memory array in a current production cycle and another brand in the next production cycle. Also, different memory arrays may be used for different storage system models priced at different levels.

[0013] Therefore, one common design approach is to have storage systems include one or more controllers with multiple specialized hardware, firmware, and/or software that encode/decode data read from or written to the memory arrays from different manufacturers, with a subset of such hardware, firmware, and/or software actually used depending on the actual memory arrays paired with the controller(s) at time of assembly or at the final design stage. As a result, storage systems can include multiple controllers or extra, unused hardware, firmware, and/or software to support multiple memory page formats for the one or more memory arrays that are not included in the finally assembled storage systems.

[0014] In the alternative, to reduce the amount of hardware, firmware, and/or software included in storage systems, storage systems and controllers can be constructed to support only one specific, known memory page format having a known memory page size and may not to support other memory page formats. Such an approach, however, limits the usefulness of storage systems and controllers to other types of memory arrays, and multiple different storage systems and controllers may need to be constructed to support multiple memory array types. In addition, in some instances, a memory page format of a memory array may not be known when storage systems and controllers are designed/constructed. Accordingly, improved systems and methods for adaptively coding data for multiple known or unknown memory storage formats are desired.

[0015] In some embodiments of the present invention, a storage system includes a controller and a non-volatile memory array having a plurality of memory pages with a common memory page size. The controller determines the common memory page size for the non-volatile memory array and selects a code word length, such as a low-density parity-check (LDPC) code word length, from multiple pre-defined code word lengths based on the common memory page size. By selectively using an appropriate code word length (chosen from multiple pre-defined code word lengths) and corresponding coding parameters, the controller uses at least some common hardware, firmware, and/or software to support multiple known or unknown memory page formats. Thus, the controller can selectively adapt its operations, so it can be paired with memory arrays of different formats made by different manufacturers.

[0016] In some embodiments, a controller of a storage system adjusts coding parameters for coding user data as memory pages, blocks, or dies of the non-volatile memory array age and/or wear out due, in part, to use of the memory array. The mechanism of adjusting coding parameters enables the controller to support additional parity per unit data over time, thereby improving the error correction or detection capabilities as the quality (e.g., data retention capabilities) of the memory array diminishes. Moreover, the mechanism of adjusting coding parameters facilitates a balancing of decoding time due to additional parity per unit data with error correction or detection benefits of additional parity data. In addition, the controller can store adjusted coding parameters for memory pages, blocks, or dies of the memory array to permit different code or parity rates for different memory pages, blocks, or dies.

System Overview

[0017] Fig. 1 illustrates a storage system 120 that adaptively codes data according to one embodiment of the invention. As is shown, a storage system 120 (e.g., hybrid hard drive, solid state drive, etc.) includes a controller 130 and a non-volatile memory array 140, which comprises one or more blocks of storage, identified as Block "A" 142 through Block "N". Each block comprises a plurality of flash pages (F-pages). For example, Block A 142 of Figure 1 includes a plurality of F-pages, identified as F-pages A 153, B, through N. In some embodiments, each "F-page" is a smallest grouping of memory cells in the non-volatile memory array 140 that can be programmed in a single operation or as a unit. Further, each F-page includes a plurality of error correcting code pages (E-pages). In the illustrated embodiment, each F-page includes four E-pages that are illustrated as four boxes, including E-page 144. Other embodiments may use F-pages or E-pages that are defined differently or each F-page may include greater or fewer than four E-pages.

[0018] The controller 130 can receive data and/or storage access commands from a storage interface module 112 (e.g., a device driver) in a host system 110. Storage access commands communicated by the storage interface 112 can include write and read commands issued by the host system 110. The commands can specify a logical block address in the storage system 120, and the controller 130 can execute the received commands in the non-volatile memory array 140. In a hybrid hard drive, data may be stored in magnetic media storage component (not shown in Fig. 1) in addition to the non-volatile memory array 140.

[0019] The storage system 120 can store data received from the host system 110 so that the storage system 120 can act as memory storage for the host system 110. To facilitate this function, the controller 130 can implement a logical interface. The logical interface can present to the host system 110 storage system memory as a set of logical addresses (e.g., contiguous address) where data can be stored. Internally, the controller 130 can map logical addresses to various physical memory addresses in the non-volatile memory array 140 and/or other memory module(s).

[0020] The controller 130 includes a coder module 132. In one embodiment, the coder module 132 determines coding parameters for decoding/encoding data (e.g., user data) read from or written to memory pages, such as E-pages, of the non-volatile memory array 140. The coding parameters can be used for decoding user data read from the non-volatile memory array 140, encoding user data for storage to the non-volatile memory array 140, and other uses such as error detection or correction. The coding parameters can include LDPC coding parameters, such as the column weight of a G or H coding matrix, the row weight of a G or H coding matrix, a P matrix size (e.g., where the P matrix is a sub-matrix of a G or H coding matrix), and the like. Further, the coder module 132 can determine parity data for unpadded or padded user data, as well as decode user data having corresponding parity data and padding. In addition, the coder module 132 can adjust a code or parity rate for coding data by adjusting the coding parameters. The controller 130 and/or coder module 132 can further include internal memory (not shown), which may be of one or more suitable memory types.

[0021] The non-volatile memory array 140 can be implemented using NAND flash memory devices. Other types of solid-state memory devices can alternatively be used, such as array of 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), NOR memory, EEPROM, Ferroelectric Memory (FeRAM), Magnetoresistive RAM (MRAM), other discrete NVM (non-volatile memory) chips, or any combination thereof. In one embodiment, the non-volatile memory array 140 preferably includes multi-level cell (MLC) devices having multi-level cells capable of storing more than a single bit of information, although single-level cell (SLC) memory devices or a combination of SLC and MLC devices may be used. In one embodiment, the storage system 120 can include other memory modules, such as one or more magnetic memory modules. The storage system 120 can further include other types of storage media, such as magnetic storage.

Adaptive Data Coding

[0022] Figure 2 is a flow diagram illustrating a process 200 of determining coding parameters for coding data according to one embodiment of the invention. The process 200 can be executed by the controller 130 and/or the coder module 132. Advantageously, the process 200 can enable the controller 130 and/or the coder module 132 to support multiple non-volatile memory page formats using a plurality of code word lengths.

[0023] At block 205, the process 200 determines a memory page size for a non-volatile memory array, such as the non-volatile memory array 140. The memory page size can be, for instance, provided by the memory array vendor or calculated based on another known memory size. For example, the memory page size can correspond to an E-page size of the non-volatile memory array, and the process 200 can calculate the E-page size by looking-up a vendor provided F-page size of the memory array and dividing the F-page size by an appropriate constant for the non-volatile memory array, such as 4 or 8. In other embodiments, the memory page size can correspond to an F-page size.

[0024] At block 210, the process 200 selects a code word length that equals or exceeds the memory page size from a plurality of code word lengths. For example, the memory page size can be 2164 bytes or octets, and the plurality of pre-defined code word lengths can include lengths of 2176 bytes and 2304 bytes. In one embodiment, the process 200 selects the LDPC code word length having a minimum size or number of bytes of data equal to or greater than the memory page size from a plurality of pre-defined LDPC code word lengths. For example, the process 200 can select the code word length of 2176 bytes, which exceeds 2164 byte memory page size and has a minimum size of the plurality of pre-defined LDPC code word lengths.

[0025] At block 215, the process 200 determines coding parameters based at least in part on the selected code word length. The coding parameters can be used when coding data read from or written to the non-volatile memory array and enable the process 200 to manage a code rate (e.g., an amount of user data per total data of a data unit, where total data includes user data and parity data) for data. In one embodiment, the coding parameters for a LDPC code word length include a column weight, a P matrix size, and a row weight, and further include at least one of a code rate, an amount of user data, and an amount of parity data.

[0026] At block 220, the process 200 stores the coding parameters. For example, the process 200 can store the coding parameters in the non-volatile memory array 140 and/or in one or more other storage media of the storage system 120. The process 200 can store the coding parameters in internal memory of the controller 130 and/or the coder module 132. The stored coding parameters can facilitate using different coding parameters for coding different pages, blocks, or other divisions and/or subdivisions of the non-volatile memory array 140 and for tracking and adjusting coding parameters over time as portions of the non-volatile memory array age and/or wear out.

[0027] Figure 3 is a table 300 illustrating relationships between LDPC coding parameters and LDPC code word lengths according to one embodiment of the invention. The table 300 can designate the LDPC code word lengths and LDPC coding parameters supported by a controller, such as the controller 130 and/or coder module 132. The table 300 can be stored in the non-volatile memory array 140, one or more other storage media of the storage system 120, and/or in the internal memory of the controller 130 and/or the coder module 132. The table 300 includes two code length columns indicating two supported LDPC code word lengths. One LDPC code word length equals 2176 bytes (2048 + 128·1 bytes), and the other LDPC code word length equals 2304 bytes (2048 + 128·2 bytes). It has been discovered that various code rates around 2 kbytes can provide an optimal trade-off between complexity and performance.

[0028] For each code length, the LDPC coding parameters of column weight, P matrix size, and row weight can be varied to enable coding of data using different designed code rates (e.g., different amounts of user data per total data of a data unit, where total data includes user data and parity data) as listed in table 300. For example, if the LDPC code word length equals 2176 bytes, the applied LDPC coding parameters can be one of the LDPC coding parameter sets at circles C1, C2, C3, or C4. In one instance, the LDPC coding parameters can be chosen at circle C3, which correspond to a column weight of 4, P matrix size of 512, and row weight of 34 and further to a code rate of 0.882 (1920 bytes of user data for 2176 byes of total code length). Further, in one embodiment, the code length, which may be defined as 2048 + 128xΔ, can be adjusted based on selecting Δ as 1, 2, etc. As is illustrated, selecting the value of Δ also affects the code rate. For example, at circle C1, selecting Δ to be 1 results in the code rate of 0.941, and if the column weight, P matrix size, and row weight remain unchanged, selecting Δ to be 2 results in the code rate of 0.944. As will be further explained, the various pre-defined code rates can be used to accommodate memory arrays of different page sizes. For example, as shown, the two code lengths of 2176 and 2304 bytes can accommodate page sizes of 2176 and 2304 bytes. It is to be noted that in practice the number of pre-defined code lengths can be much higher than two to accommodate various page sizes.

[0029] A controller supporting the LDPC coding parameters of table 300 can advantageously select one code length for a non-volatile memory array and adjust the LDPC coding parameters to code data at different code rates. For instance, when a memory array with an E-page size of 2176 bytes is relatively new (e.g., lightly used) and/or experiences or exhibits few coding errors, the LDPC coding parameters at circle C1 can be selected for coding data. The LDPC coding parameters at circle C1 correspond to a column weight of 4, P matrix size of 256, and row weight of 68 and further to a code rate of 0.941. A total of 128 bytes of parity can be used for coding 2048 bytes of data at circle C1. As memory pages, blocks, or dies of the memory array age and/or wear out, the LDPC coding parameters at circles C2, C3, and C4 can instead be selected for coding data. The controller can, accordingly, gradually increase the amount of parity per unit data from a code rate of 0.941 to codes rates of 0.926, 0.882, and 0.853 at circles C2, C3, and C4, respectively, in response to changes in the quality (e.g., loss of quality) of the non-volatile memory array.

Adaptive Code Shortening

[0030] Through adaptive code shortening, some embodiments of the invention can accommodate memory arrays with page sizes that do not match up exactly with the pre-defined code lengths. Figures 4A-4C are diagrams illustrating padded user data and parity data used in adaptive code shortening according to one embodiment of the invention. In particular, Figures 4A-4C illustrate how shortening may be used to adapt code word lengths to match the memory page size for a non-volatile memory array. Advantageously, shortening enables the controller 130 and/or coder module 132 to support a memory page format in which the memory page size does not equal any code word length of a plurality of pre-defined code word lengths. For instance, if the controller 130 and/or coder module 132 support pre-defined LDPC code word lengths equal to 2176 and 2304 bytes, shortening can be used so that the controller 130 and/or coder module 132 further support a memory page size of 2164 bytes. Moreover, shortening can permit the controller 130 and/or coder module 132 to adaptively code data to match the non-volatile memory array format without sacrificing a large bit error rate performance.

[0031] In one embodiment, shortening comprises three operations. First, padding data is added to the user data to be encoded. The padding data in one embodiment is sized to be the difference between the pre-defined code length and memory page size. Second, parity data is generated based on the padding and user data. Third, the user data and parity data, which add up to the memory page size, are stored in the memory page. The padding data is not stored, but will be appended to the user data upon decoding (e.g., when the user data and parity data is read out from the memory page at a later time).

[0032] Figure 4A illustrates padding 410 and user data 420 of a data unit 400a. The user data 420 corresponds to an amount of user data for data unit 400a, and the padding 410 corresponds to a padding that facilitates code shortening. The padding 410 can include a data set of entirely zeros, entirely ones, etc., or any known or pre-defined data pattern. Continuing the example of the previous paragraph, if the memory page size is 2164 bytes for the non-volatile memory array, the controller 130 and/or coder module 132 can select, from various pre-defined code lengths such as those shown in table 300, the shortest LDPC code word length having a size equal to or greater than 2164 bytes. In this case, the code length of 2176 bytes is selected. The controller 130 and/or coder module 132 can determine that the padding 410 should include a data set having length that is equal to the difference between the code word length (2176) and the memory page size (2164) of the non-volatile memory array, or 2176 – 2164 = 12 bytes of padding data. Depending on the LDPC coding parameters for the data unit 400a, the controller 130 and/or coder module 132 can further determine an amount of bytes reserved for user data 420 and an appropriate G coding matrix for determining and/or generating parity data.

[0033] Figure 4B illustrates an example encoding process according to one embodiment. In effect, the user data 420 is "padded" so that the combined user data and padding meet the amount of bytes reserved for user data in the chosen pre-defined code length. The parity 430 is then generated for the combined padding 410 and user data 420 of a data unit 400b. Continuing the example of the previous paragraph, the appropriate G coding matrix can be used to determine LDPC parity data for parity 430. Note that as described above, the actual distribution of the 2176 bytes between padding 410, user data 420, and parity data 430 may vary. The following table illustrates some of the possible configurations (all sizes in bytes):

Memory Page Size 2164 bytes

Configuration C1 (Figure 3)	Code Length - 2176		
	Reserved for User Data		Reserved for Parity
	2048		128
With Shortening Scheme	Padding (Figure 4 – 410)	Actual User Data (Figure 4 – 420)	Parity (Figure 4 - 430)
	12	2036	128
Memory Page Size	Not written to memory	2164 (2036 + 128)	

Memory Page Size 2164 bytes

Configuration C3 (Figure 3)	Code Length - 2176		
	Reserved for User Data		Reserved for Parity
	1920		256
With Shortening Scheme	Padding (Figure 4 – 410)	Actual User Data (Figure 4 – 420)	Parity (Figure 4 - 430)
	12	1908	256
Memory Page Size	Not written to memory	2164 (1908 + 256)	

[0034] Figure 4C illustrates a data unit 400c, having user data 420 and parity 430 of the data unit 400b with padding 410 removed. The user data 420 and the parity 430 can be written to and subsequently read from a memory page of the non-volatile memory array 140. As shown in the tables above, the amount of user data 420 and parity data 430 can equal the memory page size of the non-volatile memory array, and the padding data is not written to the page. When the page is read out later, as part of the decoding, the padding is appended back to the user data read from the page. In this manner, coding for arbitrary page sizes can be performed using one of the plurality of pre-defined code word lengths at the cost of a small loss in coding efficiency.

[0035] Figure 5 is a flow diagram illustrating a process 500 of adaptive code shortening according to one embodiment of the invention. The process 500 can be executed by the controller 130 and/or the coder module 132. Advantageously, the process 500 can enable the controller 130 and/or the coder module 132 to support memory page sizes that do not equal any code word length of a plurality of pre-defined code word lengths supported by the controller 130 and/or coder module 132. The process 500 can be used to construct and manage data units 400a, 400b, and 400c described in Figures 4A-4C.

[0036] At block 505, the process 500 receives user data. The user data can be received from the storage interface module 112 along with a write command to write the user data to a non-volatile memory array, such as the non-volatile memory array 140.

[0037] At block 510, the process 500 pads the user data with padding data. The padding data can include a data set of entirely zeros, entirely ones, or a known or pre-defined data pattern. In addition, at block 510, the process 500 can further divide the user data into units having a size equal to an amount of user per data unit, which depends on corresponding coding parameters. For example, if the non-volatile memory array has a memory page size equal to 2164 bytes and the LDPC coding parameters correspond to the parameters at circle C3 of Figure 3, the user data can be divided into units having a size equal to 1908 bytes.

[0038] At block 515, the process 500 determines parity data for the padded user data using coding parameters. Continuing the example of the previous paragraph, if the LDPC coding parameters correspond to the parameters at circle C3, an appropriate G coding matrix can be selected and used to determine LDPC parity data for the padded user data.

[0039] At block 520, the process 500 outputs the user data and parity data. For example, the process 500 can output the user data and parity data for storage to E-page 144 of F-page 143 of the non-volatile memory array 140. It can be noted that the padding described with to respect to blocks 510 and 515 can be characterized as "virtual padding" since the padding itself may not be written to the memory page.

Code Rate Adjusting

[0040] Figure 6 is a flow diagram illustrating a process 600 of adjusting a code rate according to one embodiment of the invention. The process 600 can be executed by the controller 130 and/or the coder module 132. Advantageously, the process 600 can enable the controller 130 and/or the coder module 132 to adjust the code rate (e.g., an amount of parity per unit data) of memory pages, blocks, or other divisions of a non-volatile memory array as the memory pages, blocks, or other divisions wear out and/or experience decreased quality.

[0041] At block 605, the process 600 reads user data and parity data stored in a memory page. For example, the process 600 can perform a read of F-page 143 in response to a read command from the host system 110.

[0042] At block 610, the process 600 detects a number of bit errors when decoding the user data using parity data and coding parameters. For instance, the process 600 can determine a number of detected bit errors when decoding the user data using stored parity data and LDPC coding parameters corresponding to the memory page.

[0043] At block 615, the process 600 determines whether the number of bit errors exceeds a bit error threshold. The bit error threshold can depend or vary based on the coding parameters for coding data to the memory page. For example, the bit error threshold for the LDPC coding parameters at circle C1 of Figure 3 may be lower than the bit error threshold for the LDPC coding parameters at circle C2. If the process 600 determines that the number of bit errors does not exceed the bit error threshold, the process 600 terminates. On the other hand, if the process 600 determines that the number of bit errors exceeds the bit error threshold, the process 600 moves to block 620.

[0044] At block 620, the process 600 checks whether the coding parameters can be adjusted to reduce a code rate. In other words, the process 600 can determine whether more parity data can be used for coding. In one embodiment, the process 600 can determine whether the LDPC coding parameters may be adjusted while keeping a LDPC code word length unchanged. For instance, if E-page 144 is currently coded using the LDPC coding parameters at circle C3 of Figure 3, the LDPC coding parameters can be adjusted to the parameters at circle C4. Alternatively, if E-page 144 currently is coded using LDPC coding parameters at circle C4 and the table 300 contains the only available LDPC coding parameters, the parameters may not be further adjusted to a lower code rate. If the process 600 determines that the coding parameters may not be adjusted to reduce the code rate, the process 600 terminates. On the other hand, if the process 600 determines that the coding parameters can be adjusted to reduce a code rate, the process 600 moves to block 625. In one embodiment, the change of code rate may be managed at a block level where the pages in the block are switched to a new code rate at the same time. In one embodiment, where MLC memory is used, upon a determination that a further reduced code rate cannot be used, rather than terminating the process 600, the page (or block of pages) may be configured to operate in a lower-page only mode.

[0045] At block 625, the process 600 adjusts the coding parameters and stores the adjusted coding parameters to reduce a code rate for a next write operation. The process 600 can store the adjusted coding parameters in the non-volatile memory array 140, other memory module of the storage system 120, and/or internal memory of the controller 130 and/or coder module 132. The process 600 can store an indication of the code rate or LDPC coding parameters for coding data to facilitate management of LDPC coding parameters on a memory page, block, or other level division of a non-volatile memory array. Further, the adjusted coding parameters can be used for coding user data associated with a subsequent write command received from the host system 110.

Other Variations

[0046] Those skilled in the art will appreciate that in some embodiments, other approaches and methods can be used. For example, the coding techniques disclosed herein can apply to codes besides LDPC codes, such as other iterative codes like turbo codes. In addition, although the coding parameters and other values disclosed in the table 300 of Figure 3 illustrate an example set of relationships between coding parameters and code word lengths, other or additional coding relationships can be used. The table 300 can include column weights with values less than 4 and greater than 5 (such as 3 or 6), P matrix sizes with values less than 256 bits or greater than 512 (such as 128 or 1024), Δ values of less than 1 or greater than 2 (such as -1, 0, 3, or 4), Δ values having a corresponding granularity other than 128 bytes (such as 64 bytes), and a base code length with a value of less or greater than 2048 (such as 2176). Further, an amount of parity data in each data unit can be set to different values or varied depending on a quality of a storage medium. Additionally, quality metrics other than or in addition to bit errors can be used to determine whether to adjust coding parameters for coding data. Moreover, depending on the embodiment, certain of the steps described above may be removed, and others may be added. Accordingly, the scope of the present disclosure is intended to be defined only by reference to the appended claims.

[0047] While certain embodiments have been described, these embodiments have been presented by way of example only, and are not intended to limit the scope of the protection. 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 protection. The accompanying claims and their equivalents are intended to cover such forms or modifications as would fall within the scope and spirit of the protection. For example, the systems and methods disclosed herein can be applied to hard disk drives, hybrid hard drives, and the like. In addition, other forms of storage (e.g., DRAM or SRAM, battery backed-up volatile DRAM or SRAM devices, EPROM, EEPROM memory, etc.) may additionally or alternatively be used. As another example, the various components illustrated in the figures may be implemented as software and/or firmware on a processor, ASIC/FPGA, or dedicated hardware. 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 solid-state storage system, comprising:

a non-volatile memory array comprising a plurality of memory pages, each memory page having a common memory page size; and

a controller configured to:

determine the common memory page size for the non-volatile memory array;

select a low-density parity-check (LDPC) code word length from a plurality of pre-defined LDPC code word lengths, the LDPC code word length having a size equal to or greater than the common memory page size for the non-volatile memory array; and

determine LDPC coding parameters for coding data written to or read from one or more memory pages of the non-volatile memory array based at least in part on the LDPC code word length,

wherein the controller is configured to support multiple non-volatile memory page formats using the plurality of pre-defined LDPC code word lengths and to support a memory page format in which the common memory page size does not equal any LDPC code word length of the plurality of pre-defined LDPC code word lengths.

2. The solid-state storage system of claim 1, wherein the LDPC coding parameters comprise a P matrix size, a column weight, and a row weight, and further comprise at least one of a code rate, an amount of user data, and an amount of parity data.

3. The solid-state storage system of claim 1, wherein the controller is configured to select the LDPC code word length having a minimum size equal to or greater than the common memory page size for the non-volatile memory array.

4. The solid-state storage system of claim 1, wherein the memory pages comprise error-correcting code pages (E-pages).

5. The solid-state storage system of claim 1, wherein the controller is further configured to:

determine parity data for padded user data using the LDPC coding parameters, the padded user data comprising user data and padding data; and

store the user data and the parity data in a memory page of the non-volatile memory array.

6. The solid-state storage system of claim 5, wherein an amount of the padding data of the padded user data depends at least in part on a difference between the LDPC code word length and the common memory page size for the non-volatile memory array.

7. In a data storage system comprising a controller, a method of coding data, the method comprising:

determining a common memory page size for a non-volatile memory array, the non-volatile memory array comprising a plurality of memory pages, each memory page having the common memory page size;

selecting a low-density parity-check (LDPC) code word length from a plurality of pre-defined LDPC code word lengths, the LDPC code word length having a size equal to or greater than the common memory page size for the non-volatile memory array; and

determining LDPC coding parameters for coding data written to or read from one or more memory pages of the non-volatile memory array based at least in part on the LDPC code word length.

8. The method of claim 7, wherein the LDPC coding parameters comprise a P matrix size, a column weight, and a row weight, and further comprise at least one of a code rate, an amount of user data, and an amount of parity data.

9. The method of claim 7, wherein said selecting the LDPC code word length comprises selecting the LDPC code word length having a minimum size equal to or greater than the common memory page size for the non-volatile memory array.

10. The method of claim 7, wherein the memory pages comprise error-correcting code pages (E-pages).

11. The method of claim 7, further comprising:

determining parity data for padded user data using the LDPC coding parameters, the padded user data comprising user data and padding data; and storing the user data and the parity data in a memory page of the non-volatile memory array.

12. The method of claim 11, wherein an amount of the padding data of the padded user data depends at least in part on a difference between the LDPC code word length and the common memory page size for the non-volatile memory array.

13. A solid-state storage system, comprising:

a non-volatile memory array comprising a plurality of memory pages, each memory page having a common memory page size; and

a controller configured to:

determine the common memory page size for the non-volatile memory array;

select a code word length from a plurality of pre-defined code word lengths, the code word length having a size equal to or greater than the common memory page size for the non-volatile memory array; and

determine coding parameters for coding data written to or read from one or more memory pages of the non-volatile memory array based at least in part on the code word length,

wherein the controller is configured to support multiple non-volatile memory page formats using the plurality of pre-defined code word lengths.

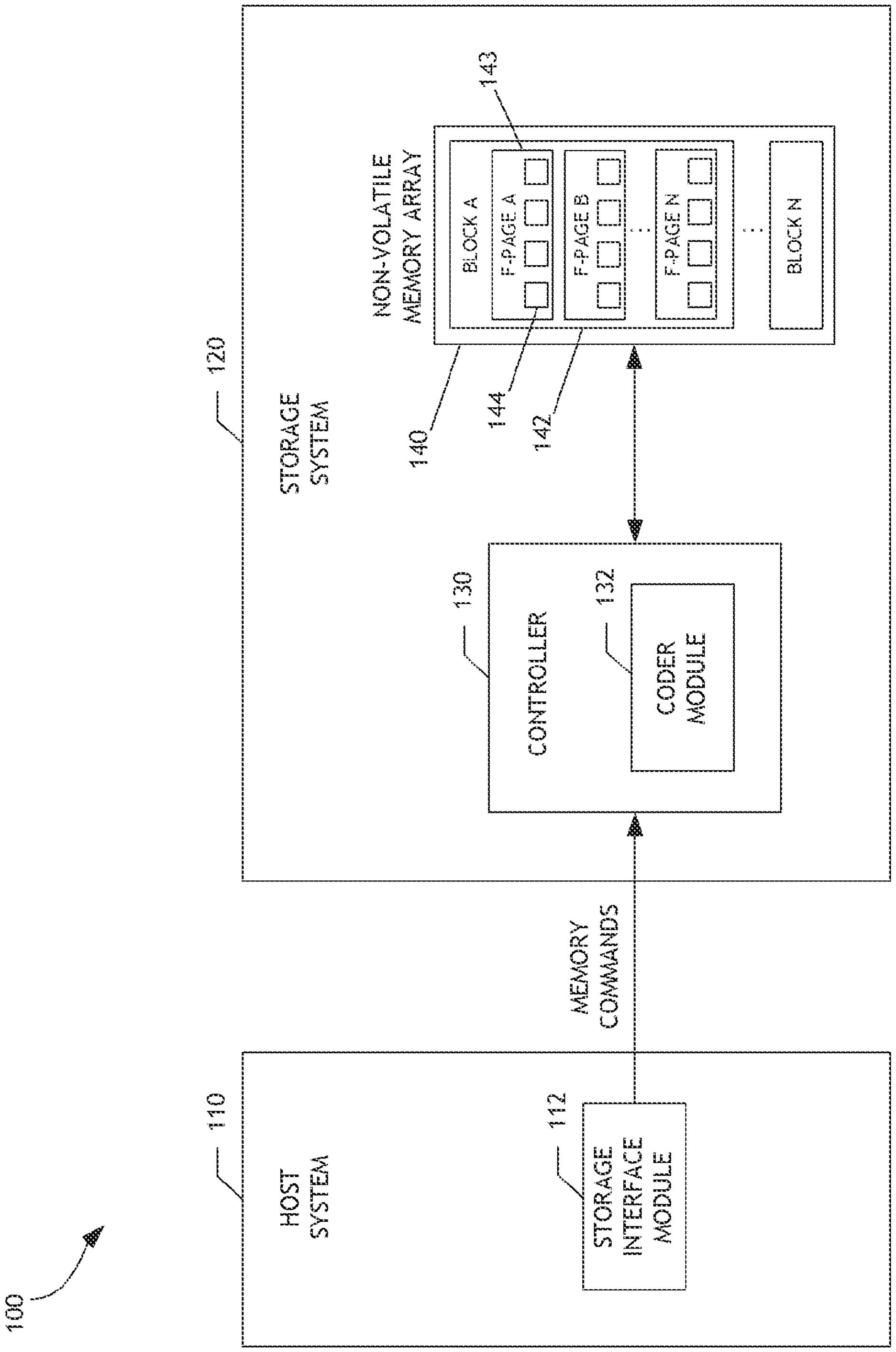
100
110
HOST SYSTEM
112
STORAGE INTERFACE MODULE
MEMORY COMMANDS
120
STORAGE SYSTEM
130
CONTROLLER
132
CODER MODULE
140
142
143
144
NON-VOLATILE MEMORY ARRAY
BLOCK A
F-PAGE A
F-PAGE B
F-PAGE N
BLOCK N

FIGURE 1

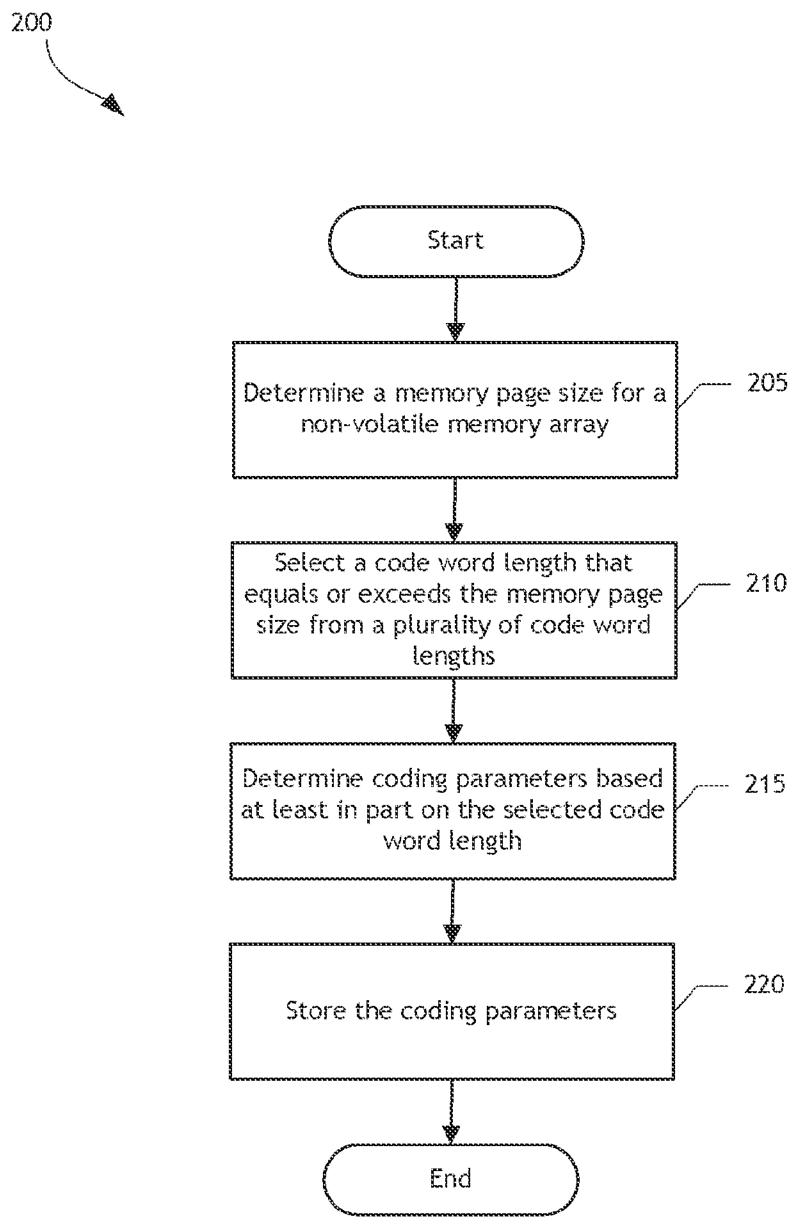
200
Start
Determine a memory page size for a non-volatile memory array
205
Select a code word length that equals or exceeds the memory page size from a plurality of code word lengths
210
Determine coding parameters based at least in part on the selected code word length
215
Store the coding parameters
220
End

FIGURE 2

300

Column Weight	P Matrix Size (bits)	Code Length = 2048 + 128·Δ (bytes)		
		Δ = 1	Δ = 2	
4	256	(C1) 68	72	Row Weight
		2176	2304	Code Length (bytes)
		2048	2176	User Data (bytes)
		0.941	0.944	Code Rate
	512	(C3) 34	36	Row Weight
		2176	2304	Code Length (bytes)
		1920	2048	User Data (bytes)
		0.882	0.889	Code Rate
5	256	(C2) 68	72	Row Weight
		2176	2304	Code Length (bytes)
		2016	2144	User Data (bytes)
		0.926	0.931	Code Rate
	512	(C4) 34	36	Row Weight
		2176	2304	Code Length (bytes)
		1856	1984	User Data (bytes)
		0.853	0.861	Code Rate

FIGURE 3

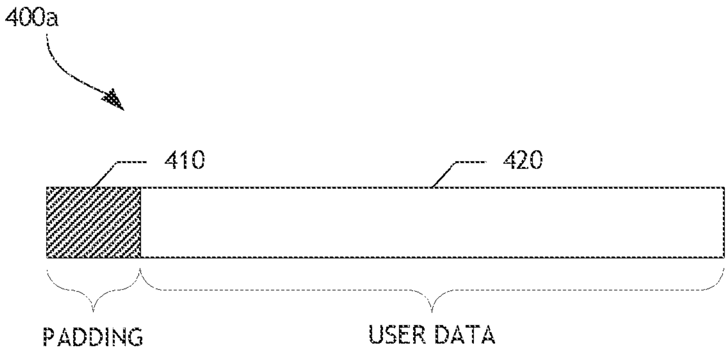

FIGURE 4A

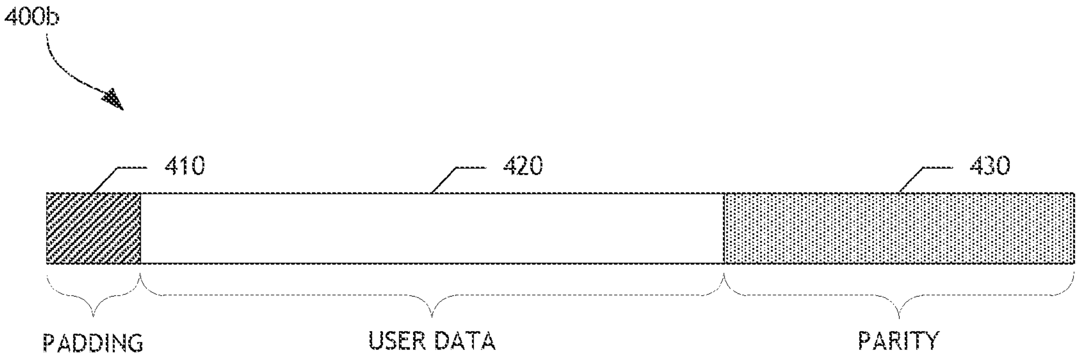

FIGURE 4B

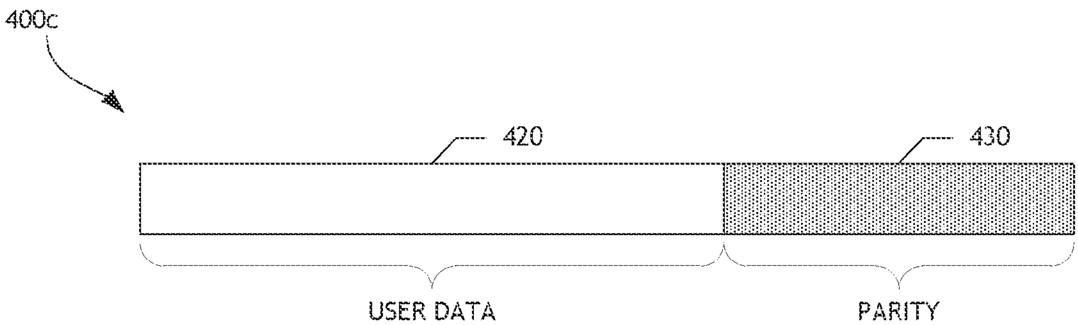

FIGURE 4C

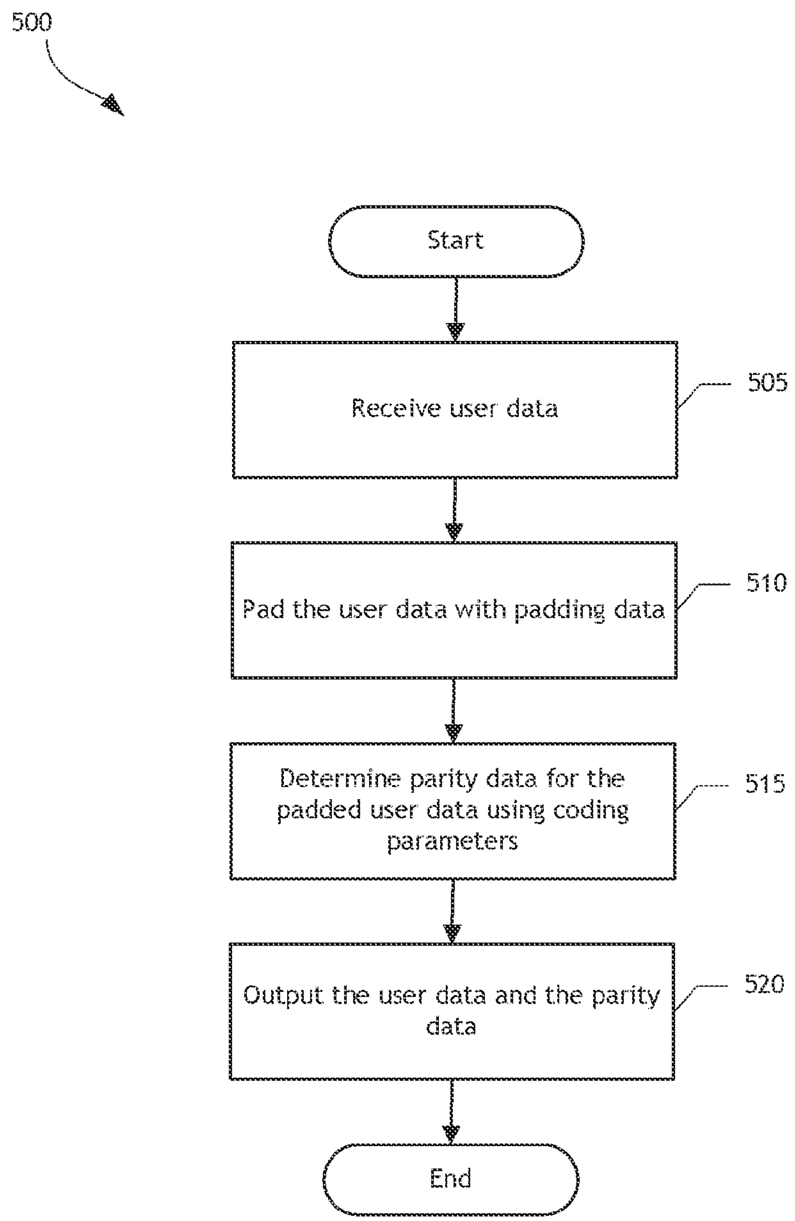
500
Start
Receive user data
505
Pad the user data with padding data
510
Determine parity data for the padded user data using coding parameters
515
Output the user data and the parity data
520
End

FIGURE 5

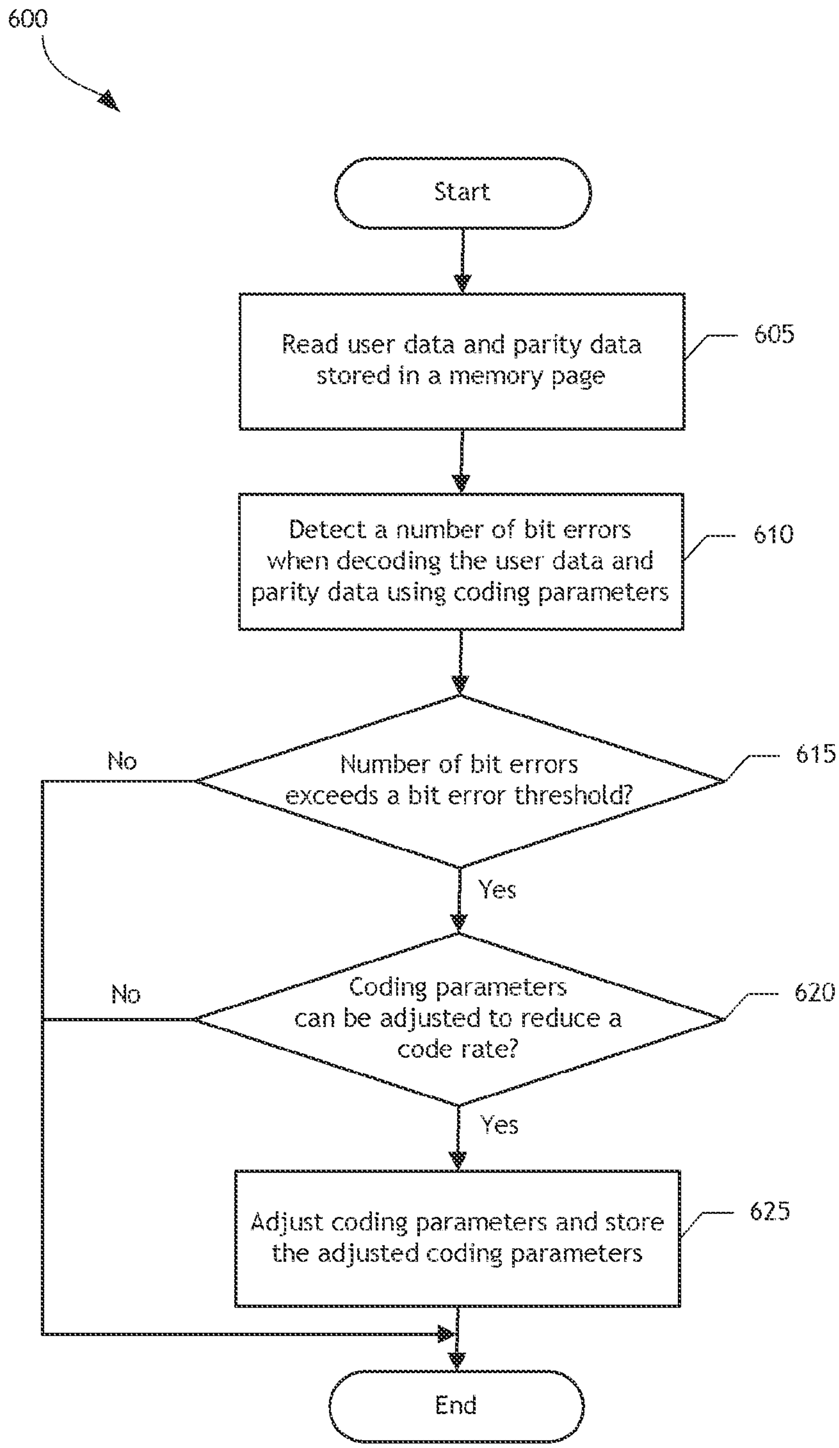
600
Start
Read user data and parity data stored in a memory page
605
Detect a number of bit errors when decoding the user data and parity data using coding parameters
610
Number of bit errors exceeds a bit error threshold?
615
No
Yes
Coding parameters can be adjusted to reduce a code rate?
620
No
Yes
Adjust coding parameters and store the adjusted coding parameters
625
End

FIGURE 6

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2013/061249

A. CLASSIFICATION OF SUBJECT MATTER

G11C 29/42(2006.01)i, G11C 16/34(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)
G11C 29/42; G06F 12/16; G11C 16/04; H03M 13/05; G06F 11/10; G06F 3/08; G11C 16/06; G11C 16/34

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords: low-density parity-check, code, word, length, error, correction, memory, parameter, format, size 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	WO 2012-058328 A1 (SANDFORCE INC.) 3 May 2012 See paragraphs 29-32, 80, 84; figure 1; and claims 1-2.	1-13
A	US 2010-0315874 A1 (RAMIN GHODSI) 16 December 2010 See paragraphs 4-5; figure 4; and claims 1, 8.	1-13
A	US 2011-0231737 A1 (KENSHI DACHIKU) 22 September 2011 See paragraphs 21-36; figure 1; and claims 1-3.	1-13
A	JP 2008-102819 A (HITACHI LTD.) 1 May 2008 See paragraphs 22-26; figures 1-2; and claim 1.	1-13
A	US 2009-0070652 A1 (SE HO MYUNG et al.) 12 March 2009 See paragraphs 70-76; figure 5; and claim 1.	1-13

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
17 January 2014 (17.01.2014)	**20 January 2014 (20.01.2014)**

Name and mailing address of the ISA/KR
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon Metropolitan City, 302-701, Republic of Korea
Facsimile No. +82-42-472-7140

Authorized officer
BYUN, Sung Cheal
Telephone No. +82-42-481-8262

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2013/061249

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
WO 2012-058328 A1	03/05/2012	CN 103329103 A	25/09/2013
		EP 2633409 A1	04/09/2013
		KR 10-2013-0096753 A	30/08/2013
		TW 201234170 A	16/08/2012
US 2010-0315874 A1	16/12/2010	CN 102804276 A	28/11/2012
		KR 10-1271912 B1	05/06/2013
		KR 10-2012-0027521 A	21/03/2012
		TW 201110124 A	16/03/2011
		US 2011-0141813 A1	16/06/2011
		US 2013-003460 A1	03/01/2013
		US 2013-242657 A1	19/09/2013
		US 7898859 B2	01/03/2011
		US 8345478 B2	01/01/2013
		US 8462552 B2	11/06/2013
		WO 2010-147770 A2	23/12/2010
		WO 2010-147770 A3	10/02/2011
US 2011-0231737 A1	22/09/2011	JP 04940322 B2	30/05/2012
		JP 2011-192206 A	29/09/2011
		KR 10-1252382 B1	08/04/2013
		KR 10-2011-0104428 A	22/09/2011
		US 8495468 B2	23/07/2013
JP 2008-102819 A	01/05/2008	EP 1914757 A2	23/04/2008
		EP 1914757 A3	01/09/2010
		JP 04932427 B2	16/05/2012
		US 2008-0098158 A1	24/04/2008
		US 2010-0241797 A1	23/09/2010
		US 7757038 B2	13/07/2010
US 2009-0070652 A1	12/03/2009	KR 10-2009-0026709 A	13/03/2009
		US 8261152 B2	04/09/2012