



- (51) **International Patent Classification:**
G06F 12/00 (2006.01) *G06F 11/10* (2006.01)
- (21) **International Application Number:**
PCT/US2013/063843
- (22) **International Filing Date:**
8 October 2013 (08.10.2013)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
13/649,822 11 October 2012 (11.10.2012) US
- (71) **Applicant:** MICRON TECHNOLOGY, INC. [US/US];
8000 S Federal Way, Boise, Idaho 83716-9632 (US).
- (72) **Inventors:** SHARIFI TEHRANI, Saeed; 7585 Charmant
Drive, Unit 812, San Diego, California 92122 (US).
RICHARDSON, Nicholas J.; 740 Archer Street, San
Diego, California 92109 (US).
- (74) **Agent:** BERGER, Mark R.; Brooks, Cameron & Huebsch
PLLC, 1221 Nicollet Avenue, Suite 500, Minneapolis,
Minnesota 55403 (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).

Published:

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

(54) **Title:** UPDATING RELIABILITY DATA

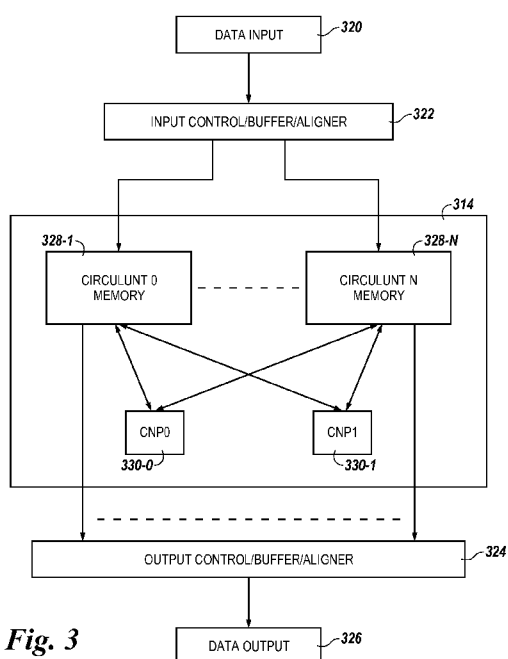


Fig. 3

(57) **Abstract:** The present disclosure includes apparatuses and methods related to updating reliability data. A number of methods can include receiving, at a variable node, either a first reliability data value with a first hard data value or a second reliability data value with a second hard data value, sending the first hard data value or the second hard data value to each check node coupled to the variable node according to a parity check code, and updating the reliability data based on input from less than all of the check nodes.

UPDATING RELIABILITY DATA

Technical Field

[0001] The present disclosure relates generally to semiconductor memory and methods, and more particularly, to apparatuses and methods related to updating reliability data.

Background

[0002] Memory devices are typically provided as internal, semiconductor, integrated circuits in computers or other electronic devices. There are many different types of memory including volatile and non-volatile memory. Volatile memory can require power to maintain its data (e.g., host data, error information, etc.) and includes random-access memory (RAM), dynamic random access memory (DRAM), and synchronous dynamic random access memory (SDRAM), among others. Non-volatile memory can provide persistent data by retaining stored data when not powered and can include NAND flash memory, NOR flash memory, read only memory (ROM), Electrically Erasable Programmable ROM (EEPROM), Erasable Programmable ROM (EPROM), and resistance variable memory such as phase change random access memory (PCRAM), resistive random access memory (RRAM), and magnetoresistive random access memory (MRAM), among others.

[0003] Memory devices can be combined together to form a storage volume of a memory system such as a solid state drive (SSD). A solid state drive can include non-volatile memory (e.g., NAND flash memory and NOR flash memory), and/or can include volatile memory (e.g., DRAM and SRAM), among various other types of non-volatile and volatile memory.

[0004] An SSD can be used to replace hard disk drives as the main storage volume for a computer, as the solid state drive can have advantages over hard drives in terms of performance, size, weight, ruggedness, operating temperature range, and power consumption. For example, SSDs can have superior performance when compared to magnetic disk drives due to their lack of moving parts, which may avoid seek time, latency, and other electro-mechanical delays associated with magnetic disk drives.

[0005] Memory is utilized as volatile and non-volatile data storage for a wide range of electronic applications. Non-volatile memory may be used in portable electronic devices, such as laptop computers, portable memory sticks, digital cameras, cellular telephones, portable music players such as MP3 players, movie players, and other electronic devices. Memory cells can be arranged into arrays, with the arrays being used in memory devices.

[0006] When data is sent (e.g., communicated, passed, transferred, transmitted, etc.) from one location to another there is the possibility that an error may occur. Errors can also occur over time while data is stored in a memory. There are a number of techniques that can be used to encode data so that an error can be detected and/or corrected. Since data is routinely sent to and from memory, and stored therein, memory can employ error correction techniques to attempt to correct data associated with the memory.

[0007] One type of error correction relies on low-density parity-check (LDPC) codes. Unencoded (e.g., “raw”) data can be encoded into codewords for transmission and/or storage. The codewords can subsequently be decoded to recover the data. Powerful error correction may be desired but balanced against latency, throughput, and/or power constraints such as those imposed by portable electronic devices.

Brief Description of the Drawings

[0008] Figure 1 is a block diagram of an apparatus in the form of a computing system including at least one memory system in accordance with a number of embodiments of the present disclosure.

[0009] Figures 2A-2I illustrate flow diagrams for updating reliability data in accordance with a number of embodiments of the present disclosure.

[0010] Figure 3 illustrates a block diagram of an apparatus including an error correction circuit in accordance with a number of embodiments of the present disclosure.

[0011] Figure 4 is a plot illustrating block error rate versus raw bit error rate (RBER) according to various approaches including at least one in accordance with a number of embodiments of the present disclosure.

Detailed Description

[0012] The present disclosure includes apparatuses and methods related to updating reliability data. A number of methods can include receiving, at a variable node, either a first reliability data value (e.g., a maximum reliability data value) with a first hard data value or a second reliability data value (e.g., a minimum reliability data value) with a second hard data value, sending the first hard data value or the second hard data value to each check node coupled to the variable node according to a parity check code, and updating the reliability data based on input from less than all of the check nodes. For example, a reliability circuit can be configured to provide specific reliability data values for particular hard data values, such as a maximum reliability data value in response to a first hard data value or a minimum reliability data value in response to a second hard data value (e.g., in hard data mode). However, in soft or semi-soft modes, the reliability circuit can be configured to provide other initial reliability data values (e.g., between a maximum and a minimum value).

[0013] In the following detailed description of the present disclosure, reference is made to the accompanying drawings that form a part hereof, and in which is shown by way of illustration how one or more embodiments of the disclosure may be practiced. These embodiments are described in sufficient detail to enable those of ordinary skill in the art to practice the embodiments of this disclosure, and it is to be understood that other embodiments may be utilized and that process, electrical, and/or structural changes may be made without departing from the scope of the present disclosure. As used herein, the designators “C”, “N”, “M”, “P”, and “V” particularly with respect to reference numerals in the drawings, indicates that a number of the particular feature so designated can be included. As used herein, “a number of” a particular thing can refer to one or more of such things (e.g., a number of memory devices can refer to one or more memory devices).

[0014] The figures herein follow a numbering convention in which the first digit or digits correspond to the drawing figure number and the remaining digits identify an element or component in the drawing. Similar elements or components between different figures may be identified by the use of similar digits. For example, 114 may reference element “14” in Figure 1, and a similar element may be referenced as 314 in Figure 3. As will be appreciated, elements

shown in the various embodiments herein can be added, exchanged, and/or eliminated so as to provide a number of additional embodiments of the present disclosure. In addition, as will be appreciated, the proportion and the relative scale of the elements provided in the figures are intended to illustrate certain embodiments of the present invention, and should not be taken in a limiting sense.

[0015] Figure 1 is a block diagram of an apparatus in the form of a computing system 100 including at least one memory system 104 in accordance with a number of embodiments of the present disclosure. As used herein, a memory system 104, a controller 108, or a memory device 110 might also be separately considered an “apparatus.” The memory system 104 can be a solid state drive (SSD), for instance, and can include a host interface 106, a controller 108 (e.g., a processor and/or other control circuitry), and a number of memory devices 110-1, . . . , 110-M (e.g., solid state memory devices such as NAND Flash devices), which provide a storage volume for the memory system 104. In another embodiment, the memory system 104 may be a single memory device.

[0016] As illustrated in Figure 1, the controller 108 can be coupled to the host interface 106 and to the memory devices 110-1, . . . , 110-M via a plurality of channels and can be used to send data between the memory system 104 and a host 102. The interface 106 can be in the form of a standardized interface. For example, when the memory system 104 is used for data storage in a computing system 100, the interface 106 can be a serial advanced technology attachment (SATA), peripheral component interconnect express (PCIe), or a universal serial bus (USB), among other connectors and interfaces. In general, however, interface 106 can provide an interface for passing control, address, data, and other signals between the memory system 104 and a host 102 having compatible receptors for the interface 106.

[0017] Host 102 can be a host system such as a personal laptop computer, a desktop computer, a digital camera, a mobile telephone, or a memory card reader, among various other types of hosts. Host 102 can include a system motherboard and/or backplane and can include a number of memory access devices (e.g., a number of processors). Host 102 can also be a memory controller, such as where memory system 104 is a memory device (e.g., having an on-die controller).

[0018] The controller 108 can communicate with the number of memory devices 110-1, . . . , 110-M (which in some embodiments can be a number of memory arrays on a single die) to control data read, write, and erase operations, among other operations. In some embodiments, the controller 108 can be on the same die or a different die than any or all of the number of memory devices 110.

[0019] Although not specifically illustrated, in some embodiments, the controller 108 can include a discrete memory channel controller for each channel coupling the controller 108 to the memory devices 110-1, . . . , 110-M. The controller 108 can include, for example, a number of components in the form of hardware and/or firmware (e.g., one or more integrated circuits) and/or software for controlling access to the number of memory devices 110-1, . . . , 110-M and/or for facilitating data transfer between the host 102 and memory devices 110-1, . . . , 110-M.

[0020] As illustrated in Figure 1, the controller 108 can include a reliability circuit 112 and an error correction circuit 114. For example, the reliability circuit 112 can be a log likelihood ratio (LLR) circuit and/or the error correction circuit 114 can be a low density parity check (LDPC) circuit. Each of the reliability circuit 112 and an error correction circuit 114 can be discrete components such as an application specific integrated circuit (ASIC) or the components may reflect functionally provided by circuitry within the controller 108 that does not necessarily have a discrete physical form separate from other portions of the controller 108. Although illustrated as components within the controller 108 in Figure 1, each of the reliability circuit 112 and an error correction circuit 114 can be external to the controller 108 or have a number of components located within the controller 108 and a number of components located external to the controller 108.

[0021] An error correction circuit 114 can include two types of processing nodes: variable nodes and check nodes. The processing nodes can be coupled according to the code's parity check code. In a number of embodiments of the present disclosure, the variable nodes can be implemented as circulant memories and the check nodes can be implemented as circuits that compute (e.g., calculate) parity data (e.g., perform a syndrome check) and/or increment and/or decrement reliability data, among other functions. As such, the check

nodes can include XOR circuits and/or up/down counters, among other circuitry. In a number of embodiments, the variable nodes can be implemented as up/down counters (e.g., saturating up/down counters) and the check nodes can be implemented as combinational logic (e.g., exclusive or (XOR) circuits). The input to the error correction circuit 114 can be hard data from the memory 110 as received from the reliability circuit 112 (e.g., included with reliability data from the reliability circuit 112). An LLR value can be stored for each variable node and the LLR value can be updated during each layer (or some multiple of the layers, or once per iteration, for example) of decoding as described herein.

[0022] In a number of embodiments, the input to the error correction circuit 114 can comprise semi-soft data or full-soft data (e.g., in response to a failure of a syndrome check on the hard data and/or the semi-soft data). Hard data is data that corresponds only to the data state of a memory cell. For example, a 2-bit memory cell can be programmed to one of four data states, where each data state corresponds to one of data 00, 01, 10, or 11. For example, the hard data (of the most significant bit (MSB) of data states 00 and 01 is 0 while the hard data of data states 10 and 11 is 1 (of the MSB). In contrast, soft data associated with a memory cell can indicate a location of a state (e.g., threshold voltage (V_t)) stored on the memory cell within a distribution of states (e.g., V_t distribution) representing the target state to which the memory cell was programmed. Additionally, soft data associated with a memory cell can indicate a probability of whether the state of the memory cell corresponds to the target state to which the memory cell was programmed. A memory device can be configured to determine a particular number of soft data bits for hard data read therefrom. For the 2-bit memory cell described above, an example of soft data could include a greater resolution that uses four bits, which could represent up to sixteen different states. Semi-soft data indicates that less than all of the soft data is sent with the hard data. Full-soft data indicates that all of the soft data is sent with the hard data. Additional functionality associated with the controller 108 is described in more detail herein.

[0023] The number of memory devices 110-1, . . . , 110-M can include a number of arrays of memory cells (e.g., non-volatile memory cells). The arrays can be Flash arrays with a NAND architecture, for example. However, embodiments are not limited to a particular type of memory array or array

architecture. The memory cells can be grouped, for instance, into a number of blocks including a number of physical pages. A number of blocks can be included in a plane of memory cells and an array can include a number of planes. As one example, a memory device may be configured to store 8KB (kilobytes) of user data per page, 128 pages of user data per block, 2048 blocks per plane, and 16 planes per device.

[0024] In operation, data can be written to and/or read from memory (e.g., memory devices 110-1, . . . , 110-M of system 104) as a page of data, for example. As such, a page of data can be referred to as a data transfer size of the memory system. Data can be sent to/from a host (e.g., host 102) in data segments referred to as sectors (e.g., host sectors). As such, a sector of data can be referred to as a data transfer size of the host.

[0025] Figures 2A-2I illustrate flow diagrams for updating reliability data in accordance with a number of embodiments of the present disclosure. Although not specifically illustrated in Figures 2A-2I, an error correction circuit implementing an LDPC scheme can include a multitude of variable nodes and check nodes coupled according to a parity check code. LDPC codes are a class of binary linear block codes in which a set of codewords spans the null space of a sparse parity check matrix H . LDPC codes can be represented by a bipartite graph called a factor graph, which illustrates the variable nodes, check nodes, and the connections therebetween. The LDPC decoding process can also be referred to as an iterative message passing process over the edges (connections between nodes) of the factor graph.

[0026] Figure 2A illustrates an initialization step where reliability data including hard data is received at a number of variable nodes 216-1, 216-2, . . . , 216-V from a reliability circuit 212 (e.g., an LLR circuit). The reliability data including hard data received by each of the variable nodes 216-1, 216-2, . . . , 216-V is not necessarily the same (e.g., variable node 216-1 can receive hard data “HD1,” variable node 216-2 can receive hard data “HD2,” and variable node 216-V can receive hard data “HD3”) as the hard data values can be derived from a codeword read from a memory device. The hard data (e.g., a codeword) can be read from a memory device (e.g., memory device 110 illustrated in Figure 1) and received by the reliability circuit 212 for determination (e.g., assignment, generation, etc.) of reliability data, which is to include the hard data. Although

not specifically illustrated, the reliability circuit 212 can provide semi-soft data and/or full-soft data to the variable nodes 216-1, 216-2, ..., 216-V (e.g., in response to a failed syndrome check for the hard data and/or the semi-soft data).

[0027] The reliability data can be generated by the reliability circuit 212. A variable node 216-1, 216-2, ..., 216-V can receive either a first reliability data value (e.g., a maximum reliability data value) with a first hard data value or a second reliability data value (e.g., a minimum reliability data value) with a second hard data value (e.g., in hard data mode). The reliability circuit 212 can be configured to provide specific reliability data values for particular hard data values, such as a maximum reliability data value in response to a first hard data value or a minimum reliability data value in response to a second hard data value (e.g., in hard data mode). However, in soft or semi-soft modes, the reliability circuit 212 can be configured to provide other initial reliability data values (e.g., between a maximum and a minimum value).

[0028] The variable nodes 216-1, 216-2, ..., 216-V can store the received reliability data, for example, a first reliability data value in response to a hard data input having a first hard data value (e.g., first binary value) or a second reliability data value in response to the hard data input having a second hard data value (e.g., second binary value). In a number of embodiments, the reliability data can be a two's complement representation of an LLR value. For example, the two's complement value can be ≥ 0 (e.g., a maximum LLR value) for "0" hard data values and the two's complement value can be < 0 (e.g., a minimum LLR value) for "1" hard data values received from the memory.

[0029] The reliability circuit 212 can be configured to provide a maximum or minimum LLR value for respective hard data values as initial inputs to the error correction circuit (e.g., error correction circuit 114 illustrated in Figure 1). In a three-bit implementation (e.g., including one bit of hard data and a two-bit count collectively representing the reliability data), the "0" hard data value can correspond to a two's complement LLR value of +3 and the "1" hard data value can correspond to a two's complement LLR value of -4. The following table illustrates a correspondence between a three-bit value that includes one bit of hard data, two bits of reliability data, and a two's complement value representing the reliability data:

Table 1

3-bit Value	Hard Data	2's Complement LLR
000	0	0
001	0	1
010	0	2
011	0	3
100	1	-4
101	1	-3
110	1	-2
111	1	-1

As illustrated in Table 1, the most significant bit (MSB) of the three-bit value can comprise the hard data value. Although not illustrated in Table 1, the two's complement LLR can be automatically set to a particular value (such as a maximum or minimum value) based on the hard data. Thus, the two's complement LLR values of 0, 1, and 2 can be initially set to 3 and the two's complement LLR values of -3, -2, and -1 can be initially set to -4. Embodiments are not limited to a three-bit implementation, the respective correspondence of hard data values to LLR values, using maximum and minimum LLR values, or using two's complement representation (e.g., as other representations such as sign-magnitude can be used) which are used to illustrate an example implementation to facilitate understanding of the present disclosure.

[0030] Figure 2B illustrates a particular variable node 216-1 and a plurality of check nodes 218-1, 218-2, 218-3,..., 218-C coupled to the particular variable node 216-1 according to a parity check code. During a first iteration, each of the variable nodes (e.g., variable nodes 216-1, 216-2,..., 216-V illustrated in Figure 2A) can send the hard data to each of the plurality of check nodes coupled thereto (e.g., variable node 216-1 can send hard data "HD1" to check nodes 218-1, 218-2, 218-3,..., 218-C). The variable node 216-1 can send the same hard data to each check node 218-1, 218-2, 218-3,..., 218-C.

[0031] Although not specifically illustrated in Figure 2B, the particular variable node 216-1 can send the stored reliability data including the hard data to the check nodes 218-1, 218-2, 218-3,..., 218-C coupled thereto. In a number of

embodiments, the particular variable node 216-1 can include an up/down counter, in which case the particular variable node 216-1 can send the hard data without the reliability data because the particular variable node 216-1 can update the reliability data. In some embodiments, the particular variable node 216-1 can be a memory without a counter, in which case the particular variable node can send the reliability data including the hard data so that the check nodes 218-1, 218-2, 218-3,..., 218-C can update the reliability data.

[0032] Figure 2C illustrates a particular check node 218-1 receiving, during the first iteration, a respective hard data value from each of a number of variable nodes 216-1, 216-2,..., 216-V that are coupled to the particular check node 218-1. As described herein, the respective hard data values received from each of the variable nodes 216-1, 216-2,..., 216-V can be independent of the other hard data values, thus Figure 2C shows the hard data as “HD1” from variable node 216-1, “HD2” from variable node 216-2, and “HD3” from variable node 216-V. The use of different designators for the hard data does not imply that the respective hard data values are different (e.g., it is possible that all of the values are the same, such as “1”s). In a number of embodiments, the particular check node 218-1 can receive a respective reliability data value with the hard data value from each of the number of variable nodes 216-1, 216-2,..., 216-V coupled thereto.

[0033] The check node 218-1 can compute parity data based at least in part (e.g., in some embodiments, based only) on the received respective hard data values (e.g., perform a syndrome check). For example, the check node 218-1 can compute first parity data by performing an XOR operation on the hard data values from each of the variable nodes 216-1, 216-2,..., 216-V. The check node 218-1 can compute second parity data by performing an XOR operation on the first parity and respective hard data received from a respective variable node (e.g., variable node 216-1). The second parity data can be sent to the respective variable node as described in more detail below. In a number of embodiments, each check node (e.g., check nodes 218-1, 218-2, 218-3,..., 218-C) can compute parity data once per clock cycle.

[0034] Figure 2D illustrates a particular check node 218-1 sending respective parity data to respective variable nodes 216-1, 216-2,..., 216-V. The parity data can be the second parity data described above (e.g., computed based

at least in part (e.g., in some embodiments, based only) on an XOR of the hard data from a respective variable node with a result of an XOR of all of the hard data received by the check node 218-1). Thus, Figure 2D illustrates the check node 218-1 sending parity data “XOR1” to variable node 216-1, sending parity data “XOR2” to variable node 216-2, and sending parity data “XOR3” to variable node 216-V. The parity data sent from the check node 218-1, as illustrated in Figure 2D can be part of a first layer of the first iteration. The layered approach to message passing is described in more detail below. The parity data can effectively replace the hard data sent from the variable nodes 216-1, 216-2,..., 216-V as updated hard data. In a number of embodiments, the particular check node 218-1 can send the respective reliability data along with the respective parity data to respective variable nodes 216-1, 216-2,..., 216-V. The respective reliability data can be the same reliability data sent by the respective variable nodes 216-1, 216-2,..., 216-V, or the reliability data can be updated by the check node 218-1 prior to transmission.

[0035] Each of the variable nodes 216-1, 216-2,..., 216-V can update their respectively stored reliability data based at least in part (e.g., in some embodiments, based only) on their respectively received parity data from the check node 218-1 and/or updated reliability data received from the check node 218-1. In a number of embodiments, the variable nodes 216-1, 216-2,..., 216-V can update their respective reliability data by incrementing or decrementing the reliability data value based at least in part (or, in some embodiments, based only) on the parity data received from the particular check node 218-1 (e.g., without regard to parity data received from any other check nodes during a past, current, or future layer of the iteration, or a different iteration of the message passing algorithm). The variable nodes 216-1, 216-2,..., 216-V can increment or decrement the stored reliability data once per clock cycle. With respect to those embodiments that include maximum and/or minimum reliability data values stored at the variable nodes 216-1, 216-2,..., 216-V, the reliability data value can be incremented and/or decremented within a range defined by the maximum and minimum values such that, for example, if the reliability data value is already at a maximum, and the received parity data value calls for the reliability data value to be incremented, it will not be incremented. In a number of embodiments, the variable nodes 216-1, 216-2,..., 216-V can update their respective reliability data

by storing the updated reliability data received from the particular check node 218-1.

[0036] Figures 2E-2H illustrate four layers of an iteration of message passing from those check nodes 218-1, 218-2, 218-3,..., 218-C coupled to a particular variable node 216-1. In a number of embodiments, one layer can include incrementing or decrementing the stored reliability data value at the variable node 216-1 based at least in part (or, in some embodiments, based only) on parity data from one of the check nodes 218-1, 218-2, 218-3,..., 218-C coupled to the particular variable node 216-1. In a number of embodiments, one layer can include updating the stored reliability data value at the variable node 216-1 based at least in part (or, in some embodiments, based only) on the updated reliability data from one of the check nodes 218-1, 218-2, 218-3,..., 218-C coupled to the particular variable node 216-1.

[0037] Figure 2E illustrates a first check node 218-1 of the check nodes 218-1, 218-2, 218-3,..., 218-C coupled to the variable node 216-1 sending parity data (e.g., "XOR-A") to the variable node 216-1 in the first layer of the first iteration. The parity data sent by the first check node 218-1 can be based at least in part (e.g., in some embodiments, based only) on hard data sent by the variable node 216-1 and other variable nodes (not illustrated). In the first layer, the variable node 216-1 can update the reliability data stored therewith based at least in part (e.g., in some embodiments, based only) on the parity data received from the first check node 218-1. Updating the reliability data in response to the parity data can include updating the parity data because of the reception of the parity data (e.g., receiving the parity data causes the reliability data to be updated). Updating the reliability data based at least in part (e.g., in some embodiments, based only) on the parity data can include incrementing or decrementing the reliability data value according to a value of the received parity data. For example, if the parity data XOR-A is a first value (e.g., binary "1"), then the variable node 216-1 can increment the reliability data and if the parity data XOR-A is a second value (e.g., binary "0"), then the variable node 216-1 can decrement the reliability data. Incrementing and/or decrementing the reliability data can include incrementing and/or decrementing the reliability data by a predetermined amount unless the incrementing or decrementing of the stored reliability data value would exceed a predetermined range for the reliability data

value. Although not specifically illustrated in Figure 2E, the check node 218-1 can send updated reliability data with the parity data to the particular variable node 216-1. In the first layer, the variable node 216-1 can update the reliability data stored therewith based at least in part (e.g., in some embodiments, based only) on the updated reliability data received from the first check node 218-1.

[0038] Figure 2F illustrates a second check node 218-2 of the check nodes 218-1, 218-2, 218-3,..., 218-C coupled to the variable node 216-1 sending parity data (e.g., "XOR-B") to the variable node 216-1 in a second layer of the first iteration. In the second layer, the variable node 216-1 can update the reliability data stored therewith based at least in part (e.g., in some embodiments, based only) on the parity data received from the second check node 218-2. Although not specifically illustrated in Figure 2F, the check node 218-2 can send updated reliability data with the parity data to the particular variable node 216-1. In the second layer, the variable node 216-1 can update the reliability data stored therewith based at least in part (e.g., in some embodiments, based only) on the updated reliability data received from the first check node 218-2.

[0039] Figure 2G illustrates a third check node 218-3 of the check nodes 218-1, 218-2, 218-3,..., 218-C coupled to the variable node 216-1 sending parity data (e.g., "XOR-C") to the variable node 216-1 in a third layer of the first iteration. In the third layer, the variable node 216-1 can update the reliability data stored therewith based at least in part (e.g., in some embodiments, based only) on the parity data received from the third check node 218-3. Although not specifically illustrated in Figure 2G, the check node 218-3 can send updated reliability data with the parity data to the particular variable node 216-1. In the third layer, the variable node 216-1 can update the reliability data stored therewith based at least in part (e.g., in some embodiments, based only) on the updated reliability data received from the first check node 218-3.

[0040] Figure 2H illustrates a fourth check node 218-C of the check nodes 218-1, 218-2, 218-3,..., 218-C coupled to the variable node 216-1 sending parity data (e.g., "XOR-D") to the variable node 216-1 in a fourth layer of the first iteration. In the fourth layer, the variable node 216-1 can update the reliability data stored therewith based at least in part (e.g., in some embodiments, based only) on the parity data received from the fourth check node 218-C. Although not specifically illustrated in Figure 2H, the check node 218-C can

send updated reliability data with the parity data to the particular variable node 216-1. In the fourth layer, the variable node 216-1 can update the reliability data stored therewith based at least in part (e.g., in some embodiments, based only) on the updated reliability data received from the first check node 218-C.

[0041] Each layer can include receiving an input from less than all (e.g., a respective one of the check nodes 218-C) of the check nodes 218-1, 218-2, 218-3,..., 218-C and updating the reliability data based at least in part (e.g., in some embodiments, based only) on the input (e.g., parity data and/or updated reliability data) from the respective one of the check nodes 218-1, 218-2, 218-3,..., 218-C. Although not specifically illustrated, some embodiments can include updating the reliability data every L-number (e.g., where L is a variable from 1 layer to all layers, where the reliability data can be updated every layer, every other layer, every third layer, and so on, up to once per iteration) of the number of layers include updating the reliability data based on the input from the respective ones of the plurality of check nodes. In a number of embodiments, the layered updating can proceed until the variable node 216-1 has received an input from each of the plurality of check nodes 218-1, 218-2, 218-3,..., 218-C. The layered incrementing or decrementing of the stored reliability data value at the variable node 216-1 for all of the number of check nodes 218-1, 218-2, 218-3,..., 218-C coupled thereto can be completed for one iteration before sending a subsequent (updated) hard data value from the variable node 216-1 to each of the check nodes 218-1, 218-2, 218-3,..., 218-C.

[0042] Figure 21 illustrates the variable node 216-1 sending updated hard data corresponding to the updated reliability data to each of the check nodes 218-1, 218-2, 218-3,..., 218-C. For example, the variable node 216-1 can send updated hard data "HD4" to check nodes 218-1, 218-2, 218-3,..., 218-C. Because the updated hard data "HD4" represents the hard data corresponding to the updated reliability data stored in association with the variable node 216-1, the same hard data value "HD4" can be sent to each of the check nodes 218-1, 218-2, 218-3,..., 218-C coupled to the variable node 216-1 after the layered updating. For example, the updated hard data can be an MSB of the updated reliability data (e.g., where the updated reliability data is an LLR). In a number of embodiments, the variable node 216-1 can send updated reliability data including the updated hard data. Sending the updated hard data from the

variable node 216-1 to the check nodes 218-1, 218-2, 218-3,..., 218-C can represent the beginning of a second iteration. The layered updating and sending of updated hard data can repeat iteratively until a particular number of iterations have been performed and/or until a syndrome check performed by the check nodes 218-1, 218-2, 218-3,..., 218-C is correct.

[0043] Figure 3 illustrates a block diagram of an apparatus including an error correction circuit 314 in accordance with a number of embodiments of the present disclosure. A data input 320 can be provided from a memory device (e.g., memory device 110-1 illustrated in Figure 1) via a reliability circuit (e.g., reliability circuit 112 illustrated in Figure 1). The data can be input to an input control/buffer/aligner 322. Although not specifically illustrated, the input control/buffer/aligner 322 can receive control and status information from control circuitry to arrange timing and signaling between the input 320 and the error correction circuit 314. The input control/buffer/aligner 322 can receive reliability data including hard data (e.g., a number of codewords) and provide the same (e.g., a circulant aligned hard data input expanded to three bits per LLR with one bit of hard data and two bits of reliability data) to a plurality of circulant memories 328-1,..., 328-N. In a number of embodiments, the input control/buffer/aligner 322 can receive semi-soft or full-soft data (e.g., in response to a failed syndrome check).

[0044] The number (N) of circulant memories 328-1,..., 328-N can be equal to a total number of circulants in an H matrix (representing the parity check code) row. A circulant memory 328-1 can store a circulant matrix that is specified by one vector that can appear in one of the columns of the circulant matrix, where the remaining columns are cyclic permutations of the vector with an offset equal to the column index. For example, an H matrix can include 20,480 columns and 2,048 rows representing the parity check code. Each variable node can have its own column in the H matrix. Each row can represent a parity check constraint and each column can represent one bit of a received codeword. The H matrix can be broken into smaller matrices called circulants (e.g., 512 x 512 matrices). Each layer of decoding can consist of a single row of circulants and each entry in the circulant can be a parity check, where each layer processes inputs from less than all (e.g., only one) of the check nodes. The

circulant memories 328-1,..., 328-N can exchange control and/or status information therebetween and/or with control circuitry.

[0045] The circulant memories 328-1,..., 328-N can be coupled to a number of check node processors 330-0,..., 330-1 according to the parity check code. The circulant memories 328-1,..., 328-N can send hard data and reliability data to the check node processors 330-0,..., 330-1. The check node processors can receive the reliability data including the hard data, compute parity data (updated hard data), update the reliability data based at least in part (e.g., in some embodiments, based only) on the parity data, and output the updated reliability data including updated hard data to the circulant memories 328-1,..., 328-N. Although only two check node processors 330-0,..., 330-1 are illustrated in Figure 3, embodiments are not so limited as more or fewer check node processors can be included with an error correction circuit 314. The check node processors 330-0,..., 330-1 can exchange control and/or status information therebetween and/or with control circuitry.

[0046] The circulant memories 328-1,..., 328-N can output updated hard data to the output control/buffer/aligner 324. In a number of embodiments, the circulant memories 328-1,..., 328-N do not output the reliability data to the output control/buffer/aligner 324. Although not specifically illustrated, the output control/buffer/aligner 324 can exchange control and/or status information with control circuitry and/or with the circulant memories 328-1,..., 328-N. The output control/buffer/aligner 324 can output the hard data to an output 326 (e.g., an output to a host).

[0047] Figure 4 is a plot illustrating block error rate versus raw bit error rate (RBER) according to various approaches including at least one in accordance with a number of embodiments of the present disclosure. The plot is comparing the block failure rate using a four bit min-sum algorithm for error correction (e.g., first curve 432 according to some previous approaches) versus using a three-bit algorithm according to a number of embodiments of the present disclosure (e.g., second curve 434) with hard data inputs. The first curve 432 is the block failure rate with four bits of reliability data including one bit of hard data as an input to the min-sum error correction algorithm. The second curve 434 is the block failure rate with three bits of reliability data including one bit of hard data as an inputs to the error correction algorithm according to a number of

embodiments of the present disclosure. The min-sum approach can have a block error rate of 9.0×10^{-6} , whereas the error correction algorithm according to a number of embodiments of the present disclosure can have a block error rate of 2.0×10^{-7} for a same RBER. Although not specifically illustrated in Figure 4, the error correcting performance can be further improved with the use of semi-soft or full-soft data.

[0048] Some advantages of a number of embodiments of the present disclosure over a four-bit min-sum approach include lower complexity error correction circuitry. For example the variable nodes can be implemented as up/down counters with associated memory (or, simply as memory when the check node provides the up/down counting functionality), and the check nodes can be implemented as combinational logic (e.g., XOR gates and/or up/down counters). Furthermore, the error correction can be implemented in a bit-serial decoding approach. Using three bits of data (hard data and reliability data) provides for 25% less memory usage than using four bits of data. Error correction circuitry according to the present disclosure can be implemented without memory resources between the variable nodes and the check nodes, for example to store output bits of the variable nodes and/or the check nodes.

Conclusion

[0049] The present disclosure includes apparatuses and methods related to updating reliability data. A number of methods can include receiving, at a variable node, either a first reliability data value with a first hard data value or a second reliability data value with a second hard data value, sending the first hard data value or the second hard data value to each check node coupled to the variable node according to a parity check code, and updating the reliability data based on input from less than all of the check nodes.

[0050] Although specific embodiments have been illustrated and described herein, those of ordinary skill in the art will appreciate that an arrangement calculated to achieve the same results can be substituted for the specific embodiments shown. This disclosure is intended to cover adaptations or variations of one or more embodiments of the present disclosure. It is to be understood that the above description has been made in an illustrative fashion, and not a restrictive one. Combination of the above embodiments, and other

embodiments not specifically described herein will be apparent to those of skill in the art upon reviewing the above description. The scope of the one or more embodiments of the present disclosure includes other applications in which the above structures and methods are used. Therefore, the scope of one or more embodiments of the present disclosure should be determined with reference to the appended claims, along with the full range of equivalents to which such claims are entitled.

[0051] In the foregoing Detailed Description, some features are grouped together in a single embodiment for the purpose of streamlining the disclosure. This method of disclosure is not to be interpreted as reflecting an intention that the disclosed embodiments of the present disclosure have to use more features than are expressly recited in each claim. Rather, as the following claims reflect, inventive subject matter lies in less than all features of a single disclosed embodiment. Thus, the following claims are hereby incorporated into the Detailed Description, with each claim standing on its own as a separate embodiment.

What is claimed is:

1. A method, comprising:
 - receiving, at a variable node, either a first reliability data value with a first hard data value or a second reliability data value with a second hard data value;
 - sending the first hard data value or the second hard data value received at the variable node to each of a plurality of check nodes coupled to the variable node according to a parity check code;
 - updating the reliability data based on input from less than all of the plurality of check nodes.
2. The method of claim 1, wherein updating the reliability data comprises updating the reliability data with the variable node.
3. The method of claim 1, wherein updating the reliability data comprises updating the reliability data at a respective one of the plurality of check nodes.
4. The method of claim 1, wherein updating the reliability data comprises layered updating of the reliability data by the variable node, wherein each layer includes:
 - receiving an input from a respective one of the plurality of check nodes; and
 - updating the reliability data based on the input from the respective one of the plurality of check nodes; and
 - wherein layered updating proceeds until the variable node has received an input from each of the plurality of check nodes.
5. The method of claim 4, wherein the method includes sending updated hard data corresponding to the updated reliability data to each of the plurality of check nodes after the layered updating.

6. The method of claim 5, wherein the updated hard data comprises a most significant bit (MSB) of the updated reliability data, wherein the reliability data comprises a log likelihood ratio (LLR).
7. The method of claim 5, wherein the method includes repeating the layered updating and sending updated hard data until a particular number of iterations have been performed.
8. The method of claim 5, wherein the method includes repeating the layered updating and sending updated hard data until a syndrome check performed by the plurality of check nodes is correct.
9. The method of any one of claims 1-8, wherein the method includes sending either the first reliability data value or the second reliability data value with the hard data to each of the plurality of check nodes.
10. A method, comprising:
 - storing a reliability data value that corresponds to a hard data value at a particular variable node;
 - sending the hard data value from the particular variable node to each of a number of check nodes coupled to the particular variable node according to a parity check code;
 - receiving a respective hard data value from each of a number of variable nodes coupled to a particular check node;
 - computing parity data based on the received respective hard data values at the particular check node; and
 - incrementing or decrementing the reliability data value based at least in part on the parity data computed by the particular check node.
11. The method of claim 10, wherein incrementing or decrementing comprises incrementing or decrementing the reliability data value based only on the parity data computed by the particular check node.

12. The method of any one of claims 10-11, wherein the method includes sending the reliability data value from the particular variable node to each of the number of check nodes coupled to the particular variable node; and
receiving a respective reliability data value from each of the number of variable nodes coupled to the particular check node; and
wherein incrementing or decrementing the stored reliability data value comprises incrementing or decrementing the reliability data value at the particular check node.

13. The method of claim 12, wherein the method includes sending the incremented or decremented reliability data value from the particular check node to the particular variable node; and
replacing the stored reliability data value with the incremented or decremented reliability data value at the particular variable node.

14. The method of any one of claims 10-11, wherein the method includes layered incrementing or decrementing of the stored reliability data value for each of the number of check nodes coupled to the particular variable node; and
wherein one layer comprises one incrementing or decrementing of the stored reliability data value based only on the parity data computed by a respective one of the number of check nodes coupled to the particular variable node.

15. The method of claim 14, wherein the method includes layered incrementing or decrementing of the stored reliability data value for all of the number of check nodes coupled to the particular variable node before sending a subsequent hard data value from the particular variable node to each of the number of check nodes coupled to the particular variable node.

16. The method of any one of claims 10-11, wherein incrementing or decrementing the stored reliability data value comprises incrementing or decrementing the stored reliability data value by a predetermined amount unless the incrementing or decrementing of the stored reliability data value would exceed a predetermined range for the reliability data value.

17. The method of claim 16, wherein the stored reliability data value comprises a log-likelihood ratio (LLR) value, and wherein storing the reliability data value comprises:
- storing a first reliability data value comprising a maximum LLR value in response to the hard data value comprising a first binary value; and
 - storing a second reliability data value comprising a minimum LLR value in response to the hard data value comprising a second binary value.
18. The method of any one of claims 10-11, wherein incrementing or decrementing the stored reliability data value comprises:
- incrementing the stored reliability data value in response to the parity data comprising a first value; and
 - decrementing the stored reliability data value in response to the parity data comprising a second value.
19. The method of claim any one of claims 10-11, wherein computing the parity data comprises:
- computing first parity data based on the received respective hard data values at the particular check node; and
 - computing second parity data based on the first parity data and the hard data value received from the particular variable node at the particular check node; and
- wherein the method includes sending the second parity data from the particular check node to the particular variable node, but not sending the second parity data to other variable nodes.
20. An apparatus, comprising:
- a plurality of check nodes; and
 - a plurality of variable nodes comprising memory that store reliability data coupled to the plurality of check nodes according to a parity check code;
- wherein the plurality of check nodes are configured to:
- receive respective hard data inputs from those of the plurality of variable nodes coupled thereto; and

compute respective parity data for those of the plurality of variable nodes coupled thereto based on the respective hard data inputs;
wherein the apparatus is configured to update respective reliability data based at least in part on the respective parity data.

21. The apparatus of claim 20, wherein the apparatus is configured to update respective reliability data at the plurality of check nodes for those of the variable nodes coupled thereto based at least in part on the respective parity data.

22. The apparatus of any one of claims 20-21, wherein the plurality of check nodes are configured to compute the respective parity data by:

computing first parity data based on the respective hard data inputs;
computing second parity data based on the first parity data and a respective hard data input from a particular variable node; and
wherein the plurality of check nodes are configured to update respective reliability data for the particular variable node based at least in part on the respective second parity data.

23. The apparatus of claim 22, wherein:
the plurality of check nodes include exclusive or (XOR) circuits;
the first parity data comprises an XOR of the respective hard data inputs;
and
the second parity data comprises an XOR of the first parity data with the respective hard data input from the particular variable node.

24. The apparatus of any one of claims 20-21, wherein the plurality of variable nodes are configured to update the reliability data stored therein based on information received from each of the check nodes coupled thereto on a separate clock cycle.

25. The apparatus of any one of claims 20-21, wherein the respective reliability data comprises a log-likelihood ratio (LLR) and the plurality of check nodes are configured to increment or decrement the LLR by a fixed amount based on the respective parity data.

26. An apparatus, comprising:
a memory device;
a reliability circuit coupled to the memory device and configured to receive hard data from the memory device and to determine reliability data assigned thereto, the reliability data comprising a first reliability data value in response to the hard data comprising a first value or comprising a second reliability data value in response to the hard data comprising a second value;
an error correction circuit coupled to the reliability circuit and configured to receive the hard data and the reliability data from the reliability circuit, wherein the error correction circuit is further configured to:
iteratively compute parity data for the hard data; and
increment or decrement the reliability data once per L-number of layers of parity data, wherein one iteration includes a plurality of layers of parity data.
27. The apparatus of claim 26, wherein the error correction circuit includes:
a plurality of check nodes configured to iteratively compute the parity data for the hard data; and
a plurality of variable nodes configured to update the reliability data.
28. The apparatus of claim 26, wherein the error correction circuit is coupled to a host and configured to provide the hard data to the host in response to either a particular number of iterations being performed or in response to a syndrome check on the hard data being correct.
29. The apparatus of any one of claims 26-28, wherein the reliability circuit comprises a log likelihood ratio (LLR) circuit and the error correction circuit comprises a low density parity check (LDPC) circuit.
30. An apparatus, comprising:
a plurality of circulant memories;
a plurality of check nodes coupled to the plurality of circulant memories according to a parity check code; and

wherein the plurality of circulant memories are each configured to provide respective hard data and respective reliability data to those of the plurality of check nodes coupled thereto; and

wherein the plurality of check nodes are each configured to compute parity data for received hard data, update received reliability data based at least in part on the parity data, and send the parity data and updated reliability data to those of the circulant memories coupled thereto.

31. The apparatus of claim 30, wherein the apparatus includes:

an input coupled to the plurality of circulant memories, the input configured to provide hard data and reliability data thereto; and

an output coupled to the plurality of circulant memories and configured to receive hard data therefrom; and

wherein the plurality of check nodes are configured to perform a syndrome check based on the hard data received from the plurality of circulant memories.

32. The apparatus of any one of claims 30-31, wherein the input is configured to provide semi-soft data to the circulant memories in response to the syndrome check failing for the hard data.

33. The apparatus of claim 32, wherein the input is configured to provide full-soft data in response to the syndrome check failing for the semi-soft data.

34. An apparatus, comprising:

a plurality of check nodes; and

a plurality of variable nodes coupled to the plurality of check nodes according to a parity check code;

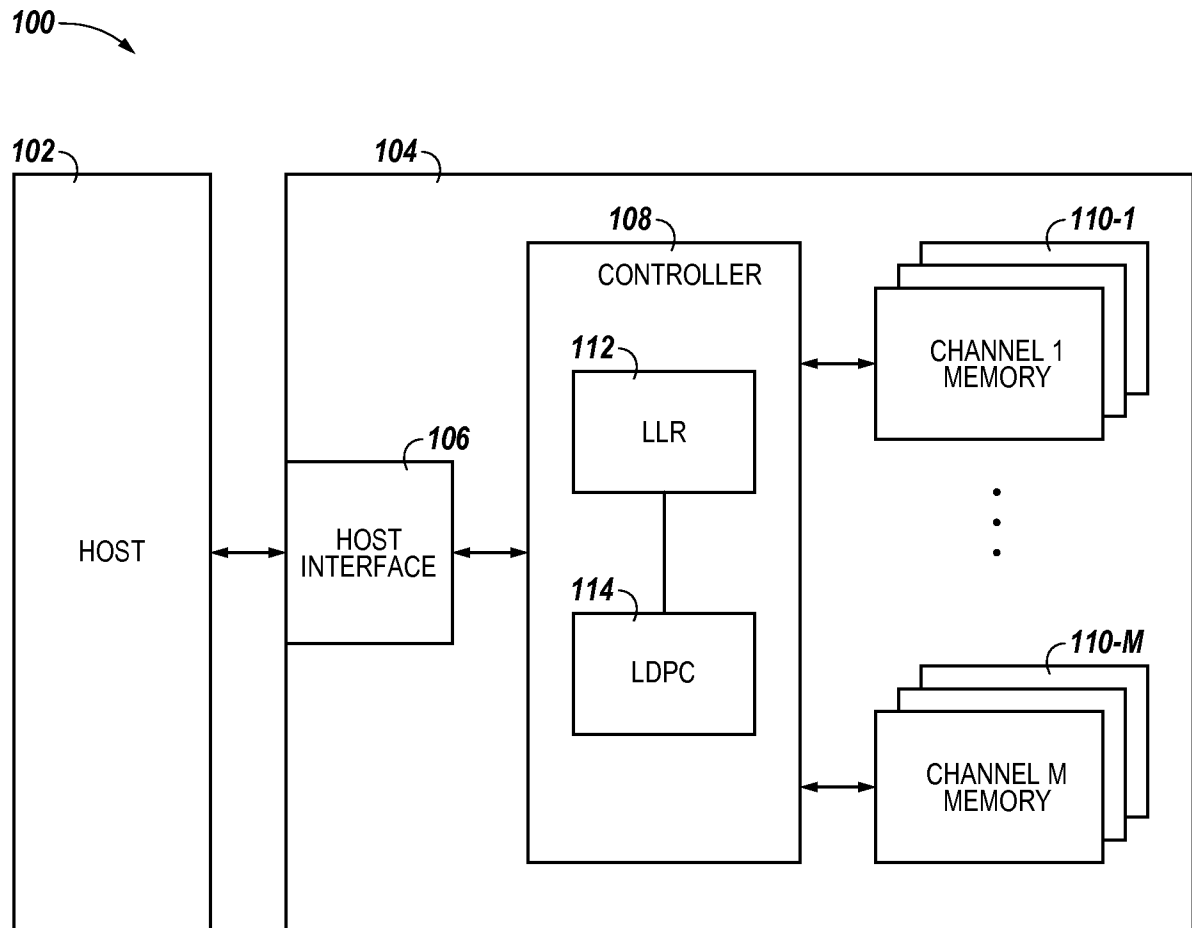
wherein the plurality of check nodes are configured to:

receive respective hard data inputs from those of the plurality of variable nodes coupled thereto; and

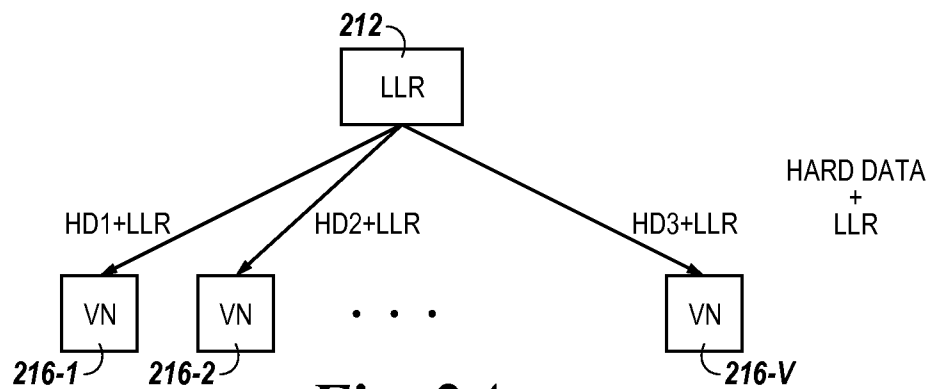
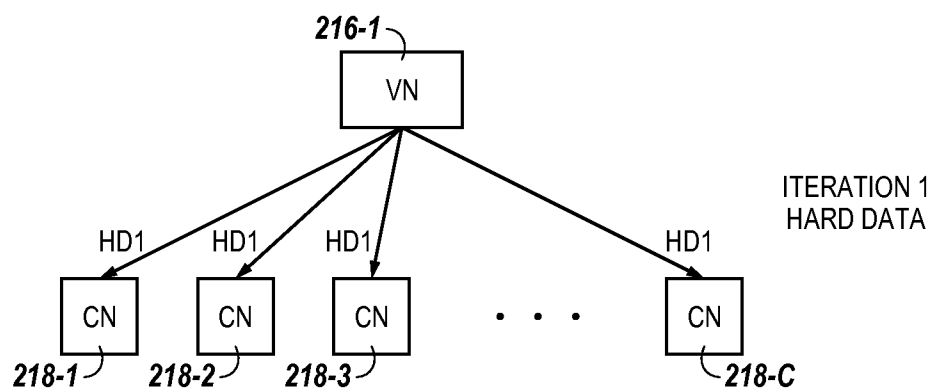
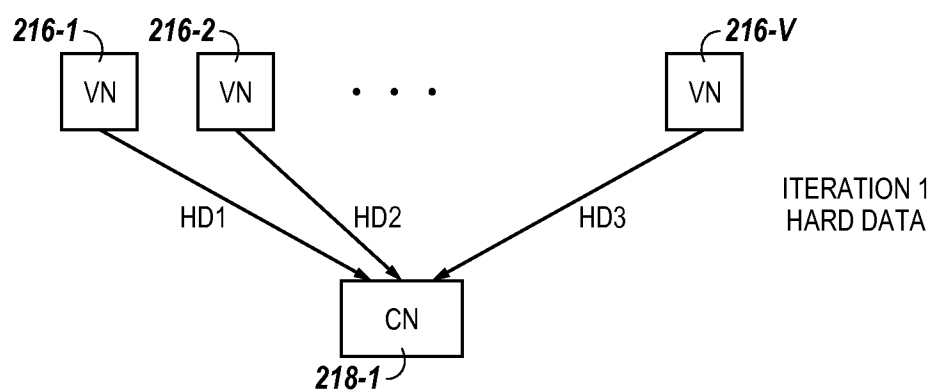
compute respective parity data for those of the plurality of variable nodes coupled thereto based on the respective hard data inputs;

wherein the plurality of variable nodes are configured to update respective reliability data based at least in part on the respective parity data.

35. The apparatus of claim 34, wherein the plurality of variable nodes comprise up/down counters.

**Fig. 1**

2/6

**Fig. 2A****Fig. 2B****Fig. 2C**

3/6

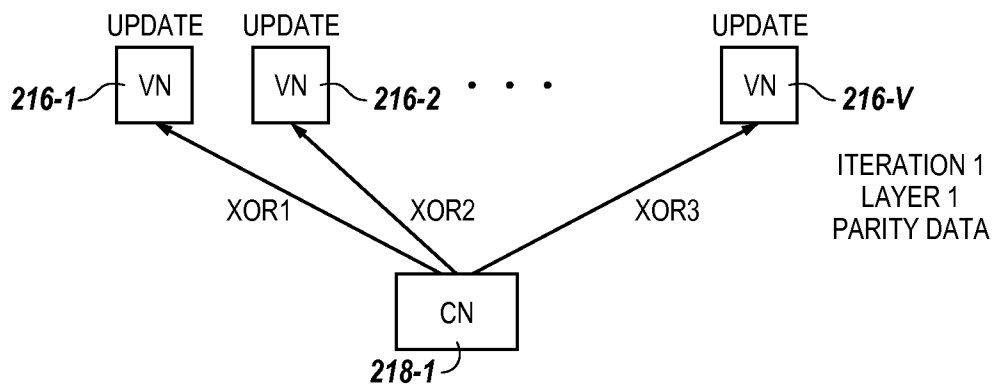


Fig. 2D

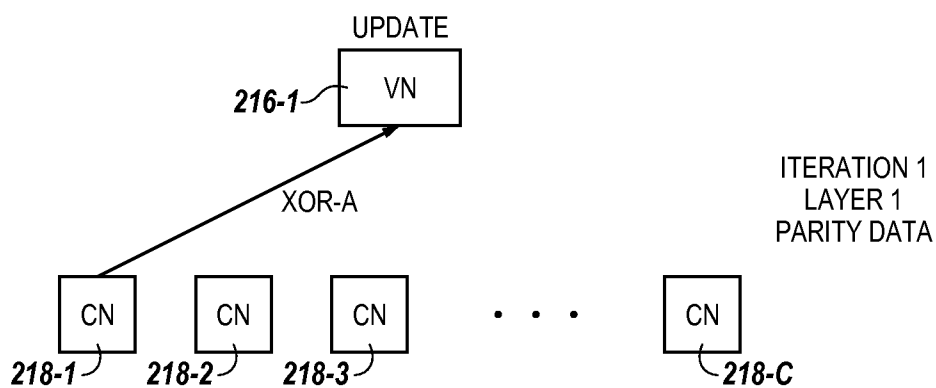


Fig. 2E

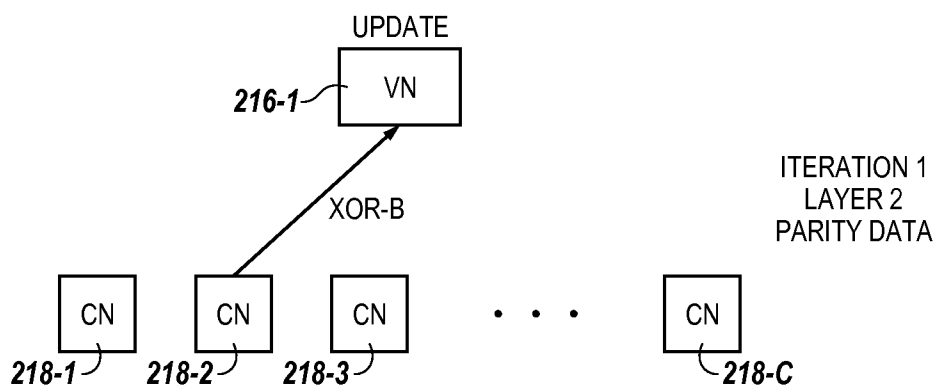


Fig. 2F

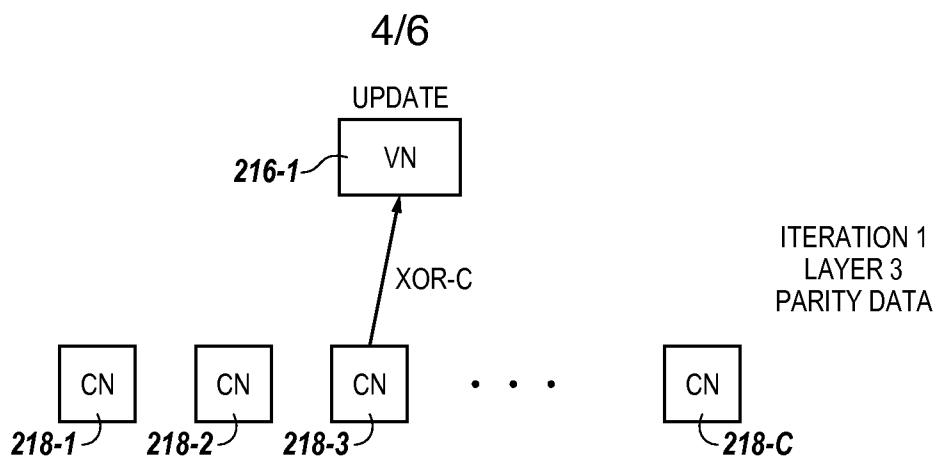


Fig. 2G

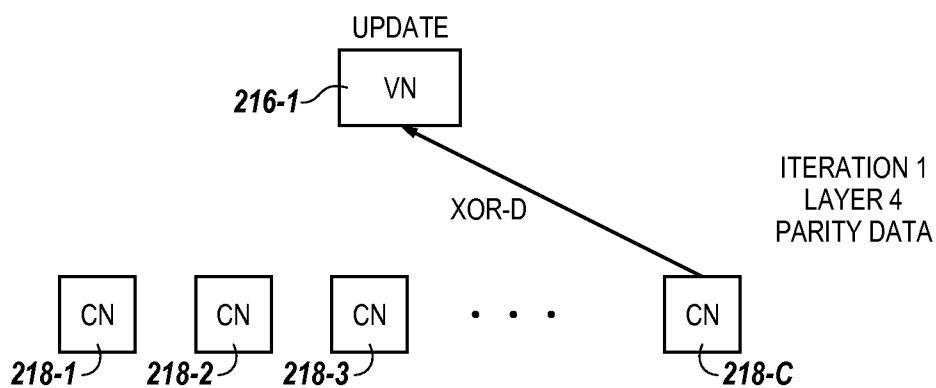


Fig. 2H

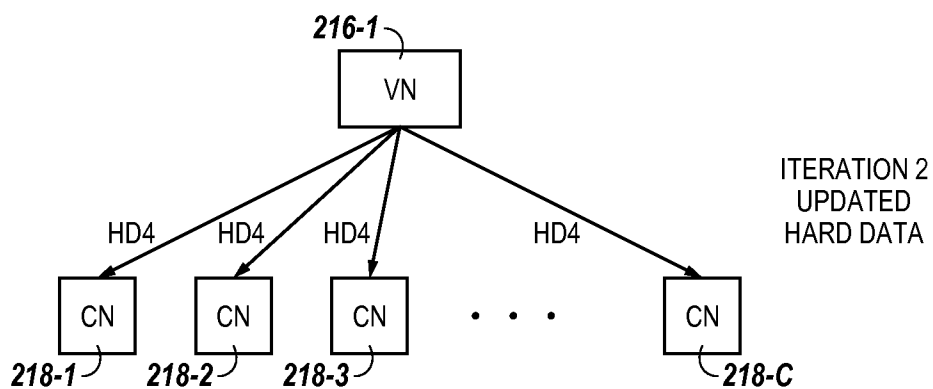
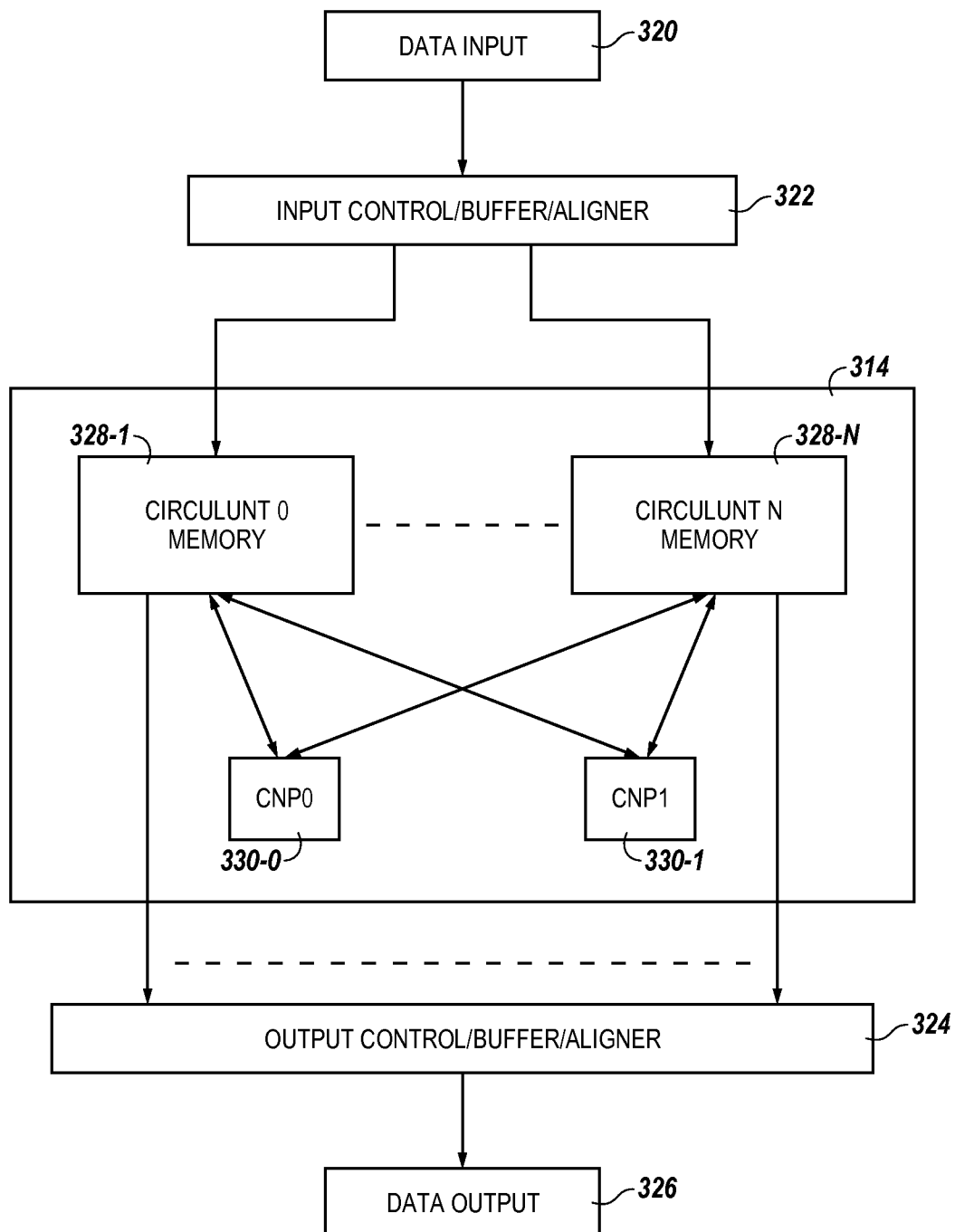
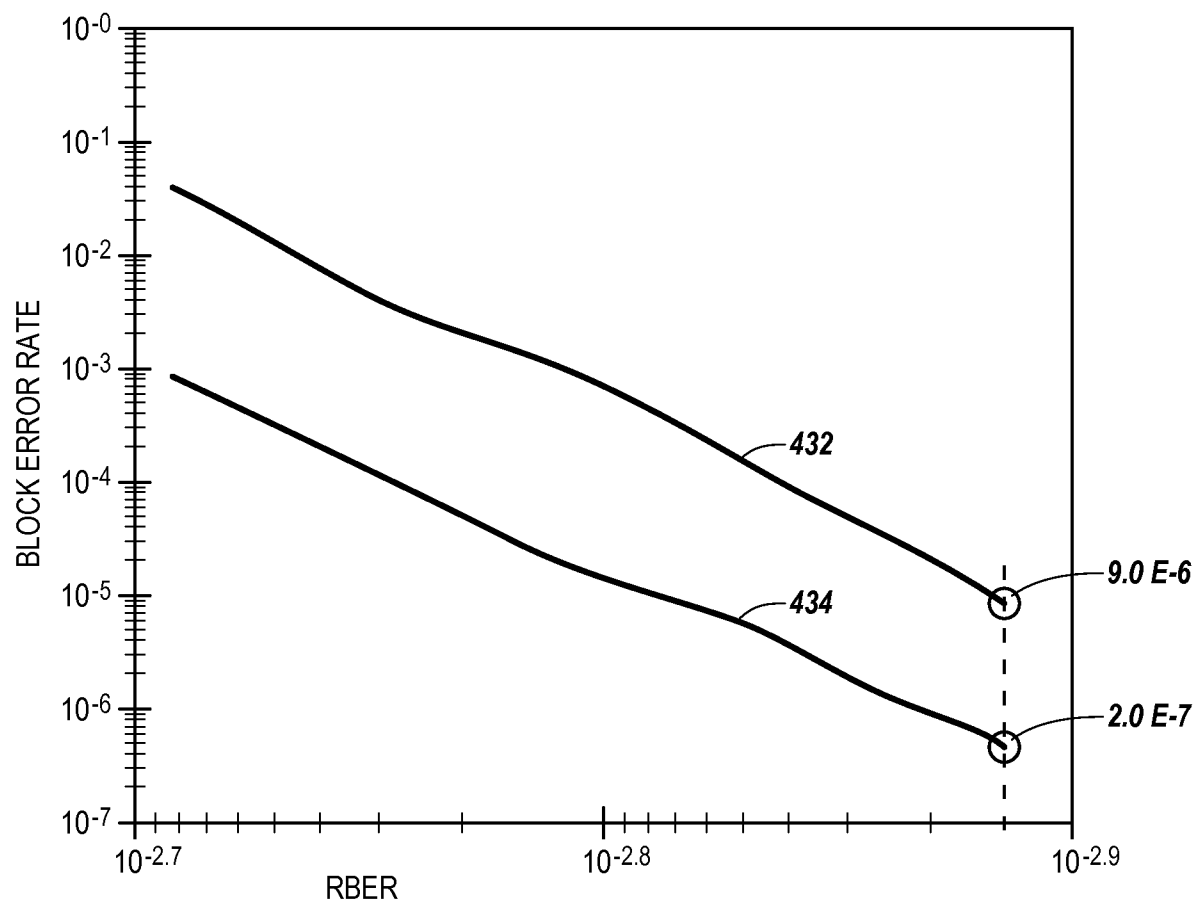


Fig. 2I

5/6

**Fig. 3**

6/6

*Fig. 4*

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2013/063843**A. CLASSIFICATION OF SUBJECT MATTER****G06F 12/00(2006.01)i, G06F 11/10(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 11/07; H03M 13/11; H03M 13/05; H03M 13/00; G06F 11/10

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: LDPC, variable node, check node, update, reliability data, 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.
Y A	US 2010-0269011 A1 (OH, JONG-EE et al.) 21 October 2010 See paragraphs [0044], [0055]-[0065], and [0082]; and figures 2, 5, 9, and 15.	1-4, 20-21, 24-25, 34 5-19, 22-23, 26-33, 35
Y A	US 2012-0246536 A1 (NG, KIN MAN et al.) 27 September 2012 See paragraph [0052]; claim 7; and figure 7.	1-4, 20-21, 24-25, 34
A	US 2011-0314352 A1 (DJORDJEVIC, IVAN B. et al.) 22 December 2011 See paragraphs [0033]-[0035] and figures 2-3.	1-35
A	US 2006-0123318 A1 (KIM, SANG-HYO et al.) 8 June 2006 See paragraphs [0007]-[0009] and [0082]; and figures 1, 3, and 7.	1-35
A	US 2010-0192036 A1 (SANDBERG MELANIE JEAN et al.) 29 July 2010 See paragraphs [0041]-[0043] and figures 2-4.	1-35



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

28 January 2014 (28.01.2014)

Date of mailing of the international search report

29 January 2014 (29.01.2014)

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

NHO, Ji Myong

Telephone No. +82-42-481-8528



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2013/063843

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2010-0269011 A1	21/10/2010	KR 10-0939334 B1 KR 10-2009-0062086 A WO 2009-075429 A1	29/01/2010 17/06/2009 18/06/2009
US 2012-0246536 A1	27/09/2012	US 2013-246880 A1 US 8219873 B1 US 8418020 B2	19/09/2013 10/07/2012 09/04/2013
US 2011-0314352 A1	22/12/2011	US 8589755 B2	19/11/2013
US 2006-0123318 A1	08/06/2006	AU 2005-225107 A1 AU 2005-225107 B2 CN 102412847 A CN 1783729 A DE 602005016355 D1 EP 1659726 A2 EP 1659726 A3 EP 1659726 B1 JP 2006-148937 A JP 4038519 B2 KR 10-0703271 B1 KR 10-2006-0057253 A US 7454685 B2	08/06/2006 14/02/2008 11/04/2012 07/06/2006 15/10/2009 24/05/2006 14/06/2006 02/09/2009 08/06/2006 30/01/2008 03/04/2007 26/05/2006 18/11/2008
US 2010-0192036 A1	29/07/2010	US 8560917 B2	15/10/2013