



- (51) International Patent Classification:
G06F 11/07 (2006.01)
- (21) International Application Number:
PCT/US2016/063425
- (22) International Filing Date:
22 November 2016 (22.11.2016)
- (25) Filing Language: English
- (26) Publication Language: English
- (30) Priority Data:
14/757,905 23 December 2015 (23.12.2015) US
- (71) Applicant: INTEL CORPORATION [US/US]; 2200 Mission College Boulevard, Santa Clara, California 95054 (US).
- (72) Inventors: DAS, Debaleena; 1511 Elwood Drive, Los Gatos, California 95032 (US). AGARWAL, Rajat; 16548 NW Talkingstick Way, Beaverton, Oregon 97006 (US). MORRIS, Brian S.; 796 E. River Pkwy, Santa Clara, California 95054 (US).
- (74) Agent: VICTOR, David W.; Konrad Raynes Davda & Victor LLP, 350 S. Beverly Dr., Suite 360, Beverly Hills, California 90212 (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, DJ, 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, KW, KZ, LA, LC, LK, LR, LS, 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, ST, 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).

[Continued on next page]

(54) Title: METHOD AND APPARATUS FOR PARTIAL CACHE LINE SPARING

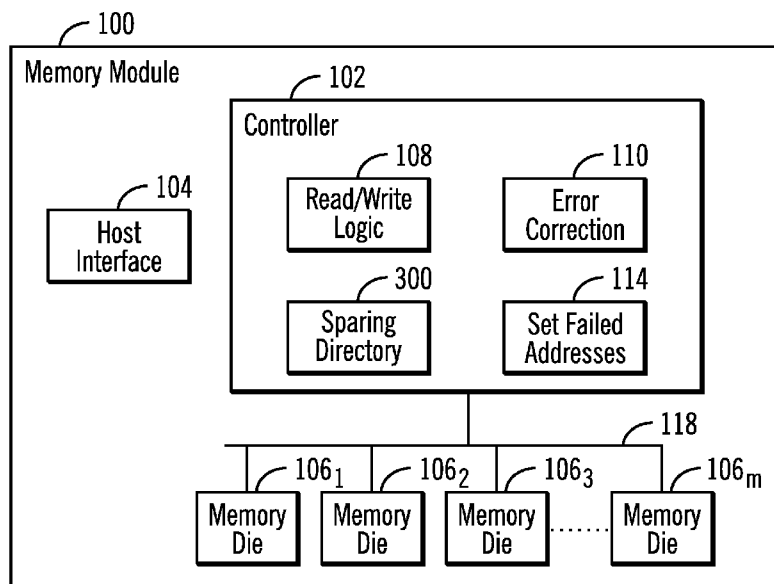


FIG. 1

(57) Abstract: Provided are an apparatus and method to store data from a cache line at locations having errors in a sparing directory. In response to a write operation having write data for locations in one of the cache lines, the write data for a location in the cache line having an error is written to an entry in a sparing directory including an address of the cache line.

WO 2017/112269 A1



Published:

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

METHOD AND APPARATUS FOR PARTIAL CACHE LINE SPARING

TECHNICAL FIELD

5 Embodiments described herein generally relate to an apparatus and method for partial cache line sparing.

BACKGROUND

 In Dynamic Random Access Memory (DRAM) systems, both hard and
10 soft errors may occur in the bits of the DRAM. Hard errors corrupt bits in a repeatable manner and concern physical damage to the memory cells whereas soft errors comprise random corrupt bits that may be corrected using an error correction algorithm and parity data for the cache line including a defective bit.

 To avoid hard errors, sparing techniques may be used to replace sections
15 of the memory dies having the errors with other spare memory dies not having errors. With row sparing, if a manufacturer or operating system during operations discovers a row of the DRAM having a defect (one or more defective bits), a fuse in the DRAM may be blown to map a spare row into the array to replace the defective row and map the defective row out of the memory array.

20 A rank sparing technique may replace a rank of DRAM devices, i.e., the physical chips, having a hard error with a spare rank on the DRAM. A memory rank is a set of DRAM chips connected to the same chip select signal that are accessed simultaneously and share command and control signals.

 There is a need in the art for improved sparing techniques.

25

BRIEF DESCRIPTION OF THE DRAWINGS

 Embodiments are described by way of example, with reference to the accompanying drawings, which are not drawn to scale, in which like reference numerals refer to similar elements.

30 FIG. 1 illustrates an embodiment of a memory module.

 FIG. 2 illustrates an embodiment of a cache line.

FIG. 3 illustrates an embodiment of a sparing directory entry.

FIG. 4 illustrates an embodiment of a read operation.

FIG. 5 illustrates an embodiment of a write operation.

FIG. 6 illustrates an embodiment of operations to perform row or rank
5 sparing.

FIG. 7 illustrates an embodiment of a system having memory modules.

DESCRIPTION OF EMBODIMENTS

Current studies indicate that around 50% of hard error failures in a
10 DRAM are single bit fails. Thus, performing row and rank sparing involves
replacing many good bits. Described embodiments provide a technique for
partial cache line sparing that performs sparing with respect to a small unit of bits
in the cache line that includes the failed single bit, such as a nibble of bits (4 bits)
in the cache line having the failed bit. In this way, the sparing focuses the
15 replacement on those specific bits experiencing the hard bit error. Thus, the
sparing only affects the cache line having the single bit failure and within the
cache line only a subset of the bits including the single bit failure. Limiting the
sparing to a small area of the cache line, such as a nibble, avoids the need to
replace large regions of the memory that do not have errors, such as occurs with
20 row and rank sparing.

With described embodiments, in response to a write operation having
write data for locations in one of the cache lines, write data for a location in the
cache line having the hard error, such as a nibble of bits or some subset of a data
unit, e.g., a byte (8 bits), is stored in an entry in a sparing directory including an
25 address of the cache line. In certain embodiments, the sparing directory may be
stored in a memory controller of the DRAM, where the controller may be
embedded in the DRAM or separate from the DRAM. In this way, one or more
failed bits may be replaced with bits that are stored in the memory controller, thus
avoiding the need to replace a larger area of the memory, such as a row or rank,
30 that may have mostly good bits.

In the following description, numerous specific details such as logic implementations, opcodes, means to specify operands, resource partitioning/sharing/duplication implementations, types and interrelationships of system components, and logic partitioning/integration choices are set forth in order to provide a more thorough understanding of the present invention. It will be appreciated, however, by one skilled in the art that the invention may be practiced without such specific details. In other instances, control structures, gate level circuits and full software instruction sequences have not been shown in detail in order not to obscure the invention. Those of ordinary skill in the art, with the included descriptions, will be able to implement appropriate functionality without undue experimentation.

References in the specification to “one embodiment,” “an embodiment,” “an example embodiment,” etc., indicate that the embodiment described may include a particular feature, structure, or characteristic, but every embodiment may not necessarily include the particular feature, structure, or characteristic. Moreover, such phrases are not necessarily referring to the same embodiment. Certain embodiments relate to storage device electronic assemblies. Embodiments include both devices and methods for forming electronic assemblies.

FIG. 1 illustrates an embodiment of a memory module 100 including a memory controller 102, a host interface 104 to interface with a host system, and a plurality of memory dies 106₁, 106₂...106_m comprising the storage elements that store data in cache lines at addressable locations. The controller 102 includes a read/write logic 108 to process read and write operations directed to cache lines in memory dies 106₁, 106₂...106_m and an error correction unit 110 to perform error correction operations using parity data and error correction codes. The controller 102 may further store a sparing directory 300 includes entries storing data for locations in the cache lines that have bit errors, such as hard bit errors. A hard bit error comprises an error which corrupts bits in a repeatable manner because of a physical defect to the memory cells, whereas a soft error comprises randomly corrupt bits in the memory that may be corrected by error correction operations using parity data. The controller 102 further maintains a set of failed addresses

114 comprising addresses of cache lines in the memory dies 106₁, 106₂...106_m that have hard bit errors for which data is stored in the sparing directory 300.

The memory controller 102 interfaces with the memory dies 106₁, 106₂...106_m over a memory interface 118.

- 5 In FIG. 1 the sparing directory 300 and set failed addresses 114 are shown as stored in the controller 102 of the memory module 100. In an alternative embodiment, the sparing directory 300 and/or the set failed addresses 114 may be stored in the memory dies 106₁, 106₂...106_m.

- The memory subsystem 100 and memory dies 106₁, 106₂...106_m may
10 implement a type of Dynamic Random Access Memories (DRAMs). In one embodiment, the memory module 100 may comprise Dual In-Line Memory Modules (DIMMs), such as unbuffered DIMMs (UDIMM), Load Reduced Dual-inline Memory Module (LRDIMM), a Small Outline Dual In-line Memory Module (SODIMM), etc. The memory module 100 may implement various
15 forms of non-volatile memory, including, but not limited to, NAND (flash) memory, ferroelectric random-access memory (FeTRAM), nanowire-based non-volatile memory, three-dimensional (3D) crosspoint memory, phase change memory (PCM), memory that incorporates memristor technology, Magnetoresistive random-access memory (MRAM), Spin Transfer Torque (STT)-
20 MRAM, etc. In one embodiment, the memory module 100 may comprise non-volatile memory devices, such as a Non-Volatile Dual In-line Memory Module (NVDIMM).

- The memory module 100 described herein may be compatible with a number of volatile memory technologies, such as DDR3 (Dual Data Rate version
25 3, original release by JEDEC (Joint Electronic Device Engineering Council) on June 27, 2007, currently on release 21), DDR4 (DDR version 4, initial specification published in September 2012 by JEDEC), LPDDR3 (low power DDR version 3, JESD209-3B, Aug 2013 by JEDEC), LPDDR4 (LOW POWER DOUBLE DATA RATE (LPDDR) version 4, JESD209-4, originally published by
30 JEDEC in August 2014), WIO2 (Wide I/O 2 (WideIO2), JESD229-2, originally published by JEDEC in August 2014), HBM (HIGH BANDWIDTH MEMORY

DRAM, JESD235, originally published by JEDEC in October 2013), DDR5 (DDR version 5, currently in discussion by JEDEC), LPDDR5 (currently in discussion by JEDEC), WIO3 (Wide I/O 3, currently in discussion by JEDEC), HBM2 (HBM version 2), currently in discussion by JEDEC), and/or others, and technologies based on derivatives or extensions of such specifications.

Although embodiments are described for a memory device, such as a DIMM, the described embodiments may be implemented in an in-package memory (IPM), such as a system-on-a chip (SOC) and other IPMs including a memory within the package.

FIG. 2 illustrates an embodiment of a cache line 200 as including a plurality of data units $202_1, 202_2, \dots, 202_n$, such as bytes or words and parity data 204 providing parity data for the data units $202_1, 202_2, \dots, 202_n$ to allow error correction of soft bit errors in the data units $202_1, 202_2, \dots, 202_n$. For instance, the data units $202_1, 202_2, \dots, 202_n$ may comprise eight bytes each to form a 64 byte cache line 200.

FIG. 3 illustrates an embodiment of an entry 300_i in the sparing directory 300 including an address 302 of the cache line 200 having the hard bit error; an offset 304 in the cache line 200 having the error; and data 306 for the location at the offset 304 having the hard error. In one embodiment, the location having the error at the offset 304 may identify a sub-unit of bits within a data unit 202_i , where the offset 304 identifies the location of the sub unit within the cache line 200. For instance, if the data units comprise bytes, than the offset 304 may indicate a location of the data 306 to store, such as a nibble, i.e., four bits, within one of the bytes. In an alternative embodiment, the sub-units stored in the data 306 of the entry 300_i may comprise other subsets of bits in a data unit 202_i , such as the entire data unit, other sub-unit of bits, or just one bit.

For instance, by way of example, if the cache line has 64 bytes, and the locations comprise nibbles within each byte, then the offset may have 7 bits to identify the offset of any one of the 128 nibbles and one bit identifies the offset of error correction code (ECC) bits 204. Thus, the offset 304 may identify all the

possible sub-units of data in the cache line 200 that may be stored in the sparing directory entry 306.

FIG. 4 illustrates an embodiment of operations performed by the read/write logic 108 to write data to an address of a cache line 200. Upon initiating (at block 400) the write operation, the read/write logic 108 detects (at block 402) an error, such as a hard error, at a location in the addressed cache line 200. The error may be detected while the read/write logic 108 is performing the write operation and detects an error when performing error correction 110. In one embodiment, the hard error in the cache line 200 may be detected by the error correction logic 110, which may detect an error while performing error correction on the cache line 200 and report the failure to firmware in the memory controller 102, which analyzes the error to determine whether there is a hard bit failure. In an alternative embodiment, the cache lines 200 having hard bit errors may have been previously determined, such as during previous operations or during manufacture and testing, and such hard bit errors are indicated in the set failed addresses 114.

Upon detecting the error at the location in the cache line 200, the read/write logic 108 adds (at block 404) an entry 300_i to the sparing directory 300 for the location in the cache line 200 indicating the address 302 and an offset 304 of the location in the cache line 200, such as the location of the nibble having the hard bit error. The read/write logic 108 may simultaneously write (at block 406) the write data to all the locations (e.g., bytes) in the cache line 200 and write data for the location, e.g., nibble, having the error to the entry 300_i in the sparing directory 300 for the location in the cache line 200. The error correction unit 110 may calculate (at block 408) parity data 204 from the write data in the write operation and store (at block 410) the parity data 204 in the cache line 200. The parity data 204 may be written while the write data is written to the data units $202_1, 202_2 \dots 202_n$.

FIG. 5 illustrates an embodiment of operations performed by the read/write logic 108 for a read operation to an address of a cache line 200 in the memory dies $106_1, 106_2 \dots 106_m$. Upon initiating (at block 500) the read operation,

if (at block 502) the read address is not indicated in the set of failed address 114, then the read/write logic 108 performs (at block 504) the read operation with respect to the addressed cache line 200, performing error correction if necessary. If (at block 502) the read address is indicated in the set failed addresses 114, then
 5 the data $202_1, 202_2, \dots, 202_n$ is read (at block 506) from the locations in the cache line 200. Substitute data 306 in the entry 300_i for the read address, as indicated in field 302, is read (at block 508) from the sparing directory entry 300_i . The data read from the location in the cache line 200 at the offset 304 indicated in the sparing directory entry 300_i , which includes the hard bit error, is replaced (at
 10 block 510) with the substitute data 306 from the entry 300_i in the sparing directory 300.

The error correction 110 may perform (at block 512) a parity check using the parity data 204 to determine whether the read data, including the substitute data 306, has an error, e.g., soft error. If (at block 514) there is an error, then the
 15 error correction 110 uses (at block 516) the parity data 204 and an error correction algorithm on the read data, including the substitute data 306, to correct the read data. If (at block 518) the error was not corrected, the read is failed (at block 520). If (at block 514) there is no error in the data or if (at block 518) the error was corrected, then the data read from the cache line 200_i with the substitute data 306
 20 for the location with the hard error is returned (at block 522) to the read operation.

FIG. 6 illustrates an embodiment of operations performed by the controller 104 to perform an additional sparing operation, such as row or rank sparing, if a hard bit error is detected when the sparing directory 300 has reached a maximum number of entries. Upon (at block 600) detecting a hard bit error in one of the
 25 cache lines 200, such as during the write operation described with respect to FIG. 4, if (at block 602) the sparing directory 300 is not full, i.e., has not reached a maximum number of entries, then the sparing directory 300 is used (at block 604) to store the data for the location having the hard bit error, such as described with respect to FIG. 4. If (at block 602) the sparing directory 300 is full, then the
 30 controller 102 performs (at block 606) one of row or rank sparing to replace the row or rank having the cache line 200 with the hard bit error with a spare row or

rank in the memory dies $106_1, 106_2 \dots 106_m$. The entries 300_i in the sparing directory 300 including address of cache lines 200 in the row or rank replaced are removed (at block 608) to make room to store data for additional hard bit errors. In this way, row or rank sparing which involves replacing large sections of the memory, are avoided until the sparing directory used to handle single hard bit errors is no longer able to handle the number of single hard bit errors being experienced.

With the described embodiments, hard bit errors in the memory device are handled by storing data in a sparing directory for a limited number of bits including the bit having the error in the cache line. When the cache line is read, the data read from the location in the cache line having the bits with the error is replaced with the data for the location that is stored in the sparing directory. Thus, the read data is reconstructed from data read from portions of the cache line not having hard errors and data maintained in the sparing data for the group of bits including the hard error. This read data with substitute data from the sparing directory may then be subject to error correction to correct any further software errors in the cache line data.

FIG. 7 illustrates an embodiment of a system 700 including a host 702 comprised of a processor 704 that executes an operating system 706 and a driver 708 to interface read and write requests from the operating system 706 to one or more memory modules $100_1, 100_2 \dots 100_n$, such as the memory module 100 of FIG. 1 described above. A host memory controller 710, in response to memory access requests from the block driver 108, communicates with the memory modules $100_1, 100_2 \dots 100_m$ over a memory bus 712, also referred to as a channel, bus interface, etc. The host memory controller 710 includes logic to manage read and write operations and memory management operations with respect to the memory modules $100_1, 100_2 \dots 100_n$ connected to the memory bus 712.

In the embodiment of FIG. 1, the controller 102, read/write logic 108, error correction 110, and sparing directory 300 are shown as included in a controller 102 within a memory module 100. In an alternative embodiment, the controller 102 may be implemented in a host memory controller 710 external to

the memory modules 100₁, 100₂...100_n, such as within host memory controller 710. In such case where the controller 102 and sparing techniques are implemented external to the memory modules 100₁, 100₂...100_n, the host memory controller 710 may manage read/write and sparing operations with respect to
5 multiple of the memory modules 100₁, 100₂...100_n.

The host 702, a Basic Input Operating System (BIOS) 714, and the host memory controller 710 may communicate over a system bus 716, such as a processor bus, fabric, ring or mesh architecture internal to the processor 704, e.g., central processing unit (CPU), Intel Corporation's KEIZER TECHNOLOGY
10 INTERCONNECT® (KTI), an Open Core Protocol interconnect, and other types of standardized or proprietary interconnects. (Keizer Technology Interconnect is a registered trademark of Intel Corporation throughout the world). Other components may also be connected to the system bus 716, such as storage devices, and input/output devices, such as display monitors, input devices, etc. The
15 memory modules 100₁, 100₂...100_n may comprise the main memory of the host 702 used to store code and operands executed by the processor 704. The BIOS 714 may perform certain hardware initialization operations of hardware in the system 700, including initializing the host memory controller 110 and memory modules 100₁, 100₂...100_n. The host memory controller 710 may be integrated
20 with the processor 104, such as a system on a chip (SOC), or implemented in logic separate from the processor 104 on the system 100 motherboard as shown in FIG. 1.

It should be appreciated that reference throughout this specification to "one embodiment" or "an embodiment" means that a particular feature, structure
25 or characteristic described in connection with the embodiment is included in at least one embodiment of the present invention. Therefore, it is emphasized and should be appreciated that two or more references to "an embodiment" or "one embodiment" or "an alternative embodiment" in various portions of this specification are not necessarily all referring to the same embodiment.
30 Furthermore, the particular features, structures or characteristics may be combined as suitable in one or more embodiments of the invention.

Similarly, it should be appreciated that in the foregoing description of embodiments of the invention, various features are sometimes grouped together in a single embodiment, figure, or description thereof for the purpose of streamlining the disclosure aiding in the understanding of one or more of the various inventive aspects. This method of disclosure, however, is not to be interpreted as reflecting an intention that the claimed subject matter requires more features than are expressly recited in each claim. Rather, as the following claims reflect, inventive aspects lie in less than all features of a single foregoing disclosed embodiment. Thus, the claims following the detailed description are hereby expressly incorporated into this detailed description.

The reference characters used herein, such as *i*, *m*, and *n* are used herein to denote a variable number of instances of an element, which may represent the same or different values, and may represent the same or different value when used with different or the same elements in different described instances.

EXAMPLES

Example 1 is an apparatus for storing data in addressable cache lines in the memory dies, comprising: a controller to perform read and write operation with respect to the memory dies, to: in response to a write operation having write data for locations in one of the cache lines, write a portion of the write data, the portion including write data for a location in the cache line having a bit error to an entry in a sparing directory in the controller, the entry including an address of the cache line.

In Example 2, the subject matter of Examples 1 and 3-12 can optionally include that the error comprises a hard error and is determined while performing error correction for the cache line.

In Example 3, the subject matter of Examples 1, 2 and 4-12 can optionally include that the controller is further to: detect the hard error while performing the write operation, wherein the writing of the data for the location in the cache line having the error is performed in response to detecting the error during the write operation.

In Example 4, the subject matter of Examples 1-3 and 5-12 can optionally include that the controller is further to: read data from the cache line having the location with the error in response to a read operation to the cache line; read substitute data for the location having the error in the cache line from the entry in the sparing directory for the cache line; replace the data with the error read at the location from the cache line with the substitute data; and returning the substitute data to the read operation.

In Example 5, the subject matter of Examples 1-4 and 6-12 can optionally include that the location in the cache line having the error comprises one of multiple locations of data in the cache line, wherein the controller is further to: write data to the locations in the cache line during the write operation; and read data for locations in the cache line, wherein the data returned to the read operation includes the data read from the cache line for locations other than the location having the error and the substitute data.

In Example 6, the subject matter of Examples 1-5 and 7-12 can optionally include that the controller is further to: determine whether the cache line is indicated as having an error in response to the read operation to the cache line, wherein the operations of reading substitute data for the location and replacing the data with the error are performed in response to determining that the cache line is indicated as having the error.

In Example 7, the subject matter of Examples 1-6 and 8-12 can optionally include that the controller is further to: perform error correction for the data read from the cache line including the substitute data in response to determining that the read data from the cache line including the substitute data includes an error; and determine whether the error correction corrected the read data from the cache line including the substitute data, wherein the data read from the cache line including the substitute data is returned in response to the error correction correcting the error.

In Example 8, the subject matter of Examples 1-7 and 9-12 can optionally include that there are multiple locations in the cache lines from which data can be stored in the sparing directory if there is an error at the locations, and wherein

each of the entries in the sparing directory includes an offset into the cache line indicating the location having the error.

In Example 9, the subject matter of Examples 1-8 and 10-12 can optionally include that each of the cache lines is comprised of a plurality of bytes, and wherein the data written to the entry in the sparing directory for the location
5 having the error comprises a subset of bits of one of the bytes in the cache line, and wherein the offset indicates the location of the subset of bits in the cache line.

In Example 10, the subject matter of Examples 1-9, 11, and 12 can optionally include that the sparing directory is within the controller and wherein
10 the apparatus includes the memory dies.

In Example 11, the subject matter of Examples 1-10 and 12 can optionally include that the sparing directory includes a plurality of entries storing data for locations in the cache lines for which errors are detected, wherein each of the entries indicates the address of the cache line having the location with the error, an
15 offset indicating the location in the cache line having the error, and the data for the location.

In Example 12, the subject matter of Examples 1-11 can optionally include that the controller is further to: determine a hard bit error in the cache line when the sparing directory includes a maximum number of entries for locations in
20 the cache lines; perform a row or rank sparing operation to replace a row or rank of the memory dies including the determined hard bit error; and remove the entries from the sparing directory including addresses for cache lines in the row or rank in the memory dies replaced with the row or rank sparing.

Example 13 is a system, comprising: a host; and a host memory
25 controller; at least one memory module coupled to the host memory controller to receive read and write requests from the host, each memory module comprising: memory dies for storing data in addressable cache lines; and a controller that performs read and write operation with respect to the memory dies, to: in
response to a write operation having write data for locations in one of the cache
30 lines, write the write data for a location in the cache line having an error to an entry in a sparing directory including an address of the cache line.

In Example 14, the subject matter of Examples 13 and 15-19 can optionally include that the controller in the memory module is further to: read data from the cache line having the location with the error in response to a read operation to the cache line; read substitute data for the location having the error in the cache line from the entry in the sparing directory for the cache line; replace the data with the error read at the location from the cache line with the substitute data; and returning the substitute data to the read operation.

In Example 15, the subject matter of Examples 13, 14, and 16-19 can optionally include that the location in the cache line having the error comprises one of multiple locations of data in the cache line, wherein the controller in the memory module is further to: write data to the locations in the cache line during the write operation; and read data for locations in the cache line, wherein the data returned to the read operation includes the data read from the cache line for locations other than the location having the error and the substitute data.

In Example 16, the subject matter of Examples 13-15 and 17-19 can optionally include that the controller in the memory module is further to: perform error correction for the data read from the cache line including the substitute data in response to determining that the read data from the cache line including the substitute data includes an error; and determine whether the error correction corrected the read data from the cache line including the substitute data, wherein the data read from the cache line including the substitute data is returned in response to the error correction correcting the error.

In Example 17, the subject matter of Examples 13-16, 18, and 19 can optionally include that there are multiple locations in the cache lines from which data can be stored in the sparing directory if there is an error at the locations, and wherein each of the entries in the sparing directory includes an offset into the cache line indicating the location having the error.

In Example 18, the subject matter of Examples 13-17, and 19 can optionally include that the sparing directory is within the controller.

In Example 19, the subject matter of Examples 13-18 can optionally include that the sparing directory includes a plurality of entries storing data for

locations in the cache lines for which errors are detected, wherein each of the entries indicates the address of the cache line having the location with the error, an offset indicating the location in the cache line having the error, and the data for the location.

5 Example 20 is a method to perform read and write operations with respect to addressable cache lines in memory dies, comprising: in response to a write operation having write data for locations in one of the cache lines, writing the write data for a location in the cache line having an error to an entry in a sparing directory including an address of the cache line.

10 In Example 21, the subject matter of Example 20 can optionally include at least any one of:

(1) that the error comprises a hard error and is determined while performing error correction for the cache line; and/or

15 (2) detecting the hard error while performing the write operation, wherein the writing of the data for the location in the cache line having the error is performed in response to detecting the error during the write operation; and/or

20 (3) reading data from the cache line having the location with the error in response to a read operation to the cache line; reading substitute data for the location having the error in the cache line from the entry in the sparing directory for the cache line; replacing the data with the error read at the location from the cache line with the substitute data; and returning the substitute data to the read operation; and/or

25 (4) that wherein the location in the cache line having the error comprises one of multiple locations of data in the cache line, writing data to the locations in the cache line during the write operation; and reading data for locations in the cache line, wherein the data returned to the read operation includes the data read from the cache line for locations other than the location having the error and the substitute data; and/or

30 (5) determining whether the cache line is indicated as having an error in response to the read operation to the cache line, wherein the operations of reading substitute data for the location and replacing the data with the error are performed

in response to determining that the cache line is indicated as having the error;
and/or

(6) performing error correction for the data read from the cache line including the substitute data in response to determining that the read data from the cache line including the substitute data includes an error; and determining whether
5 the error correction corrected the read data from the cache line including the substitute data, wherein the data read from the cache line including the substitute data is returned in response to the error correction correcting the error; and/or

(7) that there are multiple locations in the cache lines from which data can
10 be stored in the sparing directory if there is an error at the locations, and wherein each of the entries in the sparing directory includes an offset into the cache line indicating the location having the error.

(8) that each of the cache lines is comprised of a plurality of bytes, and wherein the data written to the entry in the sparing directory for the location
15 having the error comprises a subset of bits of one of the bytes in the cache line, and wherein the offset indicates the location of the subset of bits in the cache line; and

(9) that the sparing directory is within the controller and wherein the apparatus includes the memory dies; and/or

(10) that the sparing directory includes a plurality of entries storing data
20 for locations in the cache lines for which errors are detected, wherein each of the entries indicates the address of the cache line having the location with the error, an offset indicating the location in the cache line having the error, and the data for the location; and/or

(11) determining a hard bit error in the cache line when the sparing
25 directory includes a maximum number of entries for locations in the cache lines; performing a row or rank sparing operation to replace a row or rank of the memory dies including the determined hard bit error; and removing the entries from the sparing directory including addresses for cache lines in the row or rank
30 in the memory dies replaced with the row or rank sparing.

Example 22 is an apparatus, comprising: means for writing, in response to a write operation having write data for locations in one of the cache lines, the write data for a location in the cache line having an error to an entry in a sparing directory including an address of the cache line.

- 5 In Example 23, the subject matter of Example 22 can optionally include means for reading data from the cache line having the location with the error in response to a read operation to the cache line; means for reading substitute data for the location having the error in the cache line from the entry in the sparing directory for the cache line; means for replacing the data with the error read at the
10 location from the cache line with the substitute data; and means for returning the substitute data to the read operation.

Example 24 is a machine-readable storage including machine-readable instructions, when executed, to implement a method or realize an apparatus or system as claimed in any preceding claim.

- 15 Example 25 is an apparatus comprising means to perform a method as claimed in any preceding claim.

WHAT IS CLAIMED

1. An apparatus for storing data in addressable cache lines in the memory dies, comprising:

a controller to perform read and write operation with respect to the memory dies, to:

5 in response to a write operation having write data for locations in one of the cache lines, write a portion of the write data, the portion including write data for a location in the cache line having a bit error to an entry in a sparing directory in the controller, the entry including an address of the cache line.

10

2. The apparatus of claim 1, wherein the error comprises a hard error and is determined while performing error correction for the cache line.

3. The apparatus of claim 2, wherein the controller is further to:
15 detect the hard error while performing the write operation, wherein the writing of the data for the location in the cache line having the error is performed in response to detecting the error during the write operation.

4. The apparatus of any one of claims 1, 2, and 3, wherein the
20 controller is further to:
read data from the cache line having the location with the error in response to a read operation to the cache line;
read substitute data for the location having the error in the cache line from the entry in the sparing directory for the cache line;
25 replace the data with the error read at the location from the cache line with the substitute data; and
returning the substitute data to the read operation.

5. The apparatus of claim 4, wherein the location in the cache line having the error comprises one of multiple locations of data in the cache line, wherein the controller is further to:

- write data to the locations in the cache line during the write operation; and
5 read data for locations in the cache line, wherein the data returned to the read operation includes the data read from the cache line for locations other than the location having the error and the substitute data.

6. The apparatus of claim 4, wherein the controller is further to:
10 determine whether the cache line is indicated as having an error in response to the read operation to the cache line, wherein the operations of reading substitute data for the location and replacing the data with the error are performed in response to determining that the cache line is indicated as having the error.

- 15 7. The apparatus of claim 4, wherein the controller is further to:
perform error correction for the data read from the cache line including the substitute data in response to determining that the read data from the cache line including the substitute data includes an error; and
determine whether the error correction corrected the read data from the
20 cache line including the substitute data, wherein the data read from the cache line including the substitute data is returned in response to the error correction correcting the error.

8. The apparatus of any one of claims 1, 2, 3, and 4, wherein there are
25 multiple locations in the cache lines from which data can be stored in the sparing directory if there is an error at the locations, and wherein each of the entries in the sparing directory includes an offset into the cache line indicating the location having the error.

- 30 9. The apparatus of claim 8, wherein each of the cache lines is comprised of a plurality of bytes, and wherein the data written to the entry in the

sparing directory for the location having the error comprises a subset of bits of one of the bytes in the cache line, and wherein the offset indicates the location of the subset of bits in the cache line.

5 10. The apparatus of any one of claims 1, 2, and 3, wherein the sparing directory is within the controller and wherein the apparatus includes the memory dies.

10 11. The apparatus of any one of claims 1, 2, and 3, wherein the sparing directory includes a plurality of entries storing data for locations in the cache lines for which errors are detected, wherein each of the entries indicates the address of the cache line having the location with the error, an offset indicating the location in the cache line having the error, and the data for the location.

15 12. The apparatus of any one of claim 1, 2, and 3, wherein the controller is further to:

 determine a hard bit error in the cache line when the sparing directory includes a maximum number of entries for locations in the cache lines;

20 perform a row or rank sparing operation to replace a row or rank of the memory dies including the determined hard bit error; and

 remove the entries from the sparing directory including addresses for cache lines in the row or rank in the memory dies replaced with the row or rank sparing.

25 13. A system, comprising:

 a host; and

 a host memory controller;

 at least one memory module coupled to the host memory controller to receive read and write requests from the host, each memory module comprising:

30 memory dies for storing data in addressable cache lines; and

a controller that performs read and write operation with respect to the memory dies, to:

5 in response to a write operation having write data for locations in one of the cache lines, write the write data for a location in the cache line having an error to an entry in a sparing directory including an address of the cache line.

14. The system of claim 13, wherein the controller in the memory module is further to:

10 read data from the cache line having the location with the error in response to a read operation to the cache line;
 read substitute data for the location having the error in the cache line from the entry in the sparing directory for the cache line;
 replace the data with the error read at the location from the cache line with
15 the substitute data; and
 returning the substitute data to the read operation.

15. The system of claim 14, wherein the location in the cache line having the error comprises one of multiple locations of data in the cache line,
20 wherein the controller in the memory module is further to:
 write data to the locations in the cache line during the write operation; and
 read data for locations in the cache line, wherein the data returned to the read operation includes the data read from the cache line for locations other than the location having the error and the substitute data.

25

16. The system of claim 15, wherein the controller in the memory module is further to:

 perform error correction for the data read from the cache line including the substitute data in response to determining that the read data from the cache line
30 including the substitute data includes an error; and

determine whether the error correction corrected the read data from the cache line including the substitute data, wherein the data read from the cache line including the substitute data is returned in response to the error correction correcting the error.

5

17. The system of any one of claims 13-16, wherein there are multiple locations in the cache lines from which data can be stored in the sparing directory if there is an error at the locations, and wherein each of the entries in the sparing directory includes an offset into the cache line indicating the location having the error.

10

18. The system of any one of claims 13-16, wherein the sparing directory is within the controller.

15

19. A method to perform read and write operations with respect to addressable cache lines in memory dies, comprising:

in response to a write operation having write data for locations in one of the cache lines, writing the write data for a location in the cache line having an error to an entry in a sparing directory including an address of the cache line.

20

20. The method of claim 19, further comprising:

reading data from the cache line having the location with the error in response to a read operation to the cache line;

reading substitute data for the location having the error in the cache line from the entry in the sparing directory for the cache line;

25

replacing the data with the error read at the location from the cache line with the substitute data; and

returning the substitute data to the read operation.

21. The method of claim 20, wherein the location in the cache line having the error comprises one of multiple locations of data in the cache line, further comprising:

writing data to the locations in the cache line during the write operation;

5 and

reading data for locations in the cache line, wherein the data returned to the read operation includes the data read from the cache line for locations other than the location having the error and the substitute data.

10 22. The method of claim 21, further comprising:

performing error correction for the data read from the cache line including the substitute data in response to determining that the read data from the cache line including the substitute data includes an error; and

15 determining whether the error correction corrected the read data from the cache line including the substitute data, wherein the data read from the cache line including the substitute data is returned in response to the error correction correcting the error.

20 23. The method of any one of claims 19-22, wherein there are multiple locations in the cache lines from which data can be stored in the sparing directory if there is an error at the locations, and wherein each of the entries in the sparing directory includes an offset into the cache line indicating the location having the error.

25 24. An apparatus, comprising:

means for writing, in response to a write operation having write data for locations in one of the cache lines, the write data for a location in the cache line having an error to an entry in a sparing directory including an address of the cache line.

30

25. The apparatus of claim 24, further comprising:
- means for reading data from the cache line having the location with the error in response to a read operation to the cache line;
 - means for reading substitute data for the location having the error in the
 - 5 cache line from the entry in the sparing directory for the cache line;
 - means for replacing the data with the error read at the location from the cache line with the substitute data; and
 - means for returning the substitute data to the read operation.

1/5

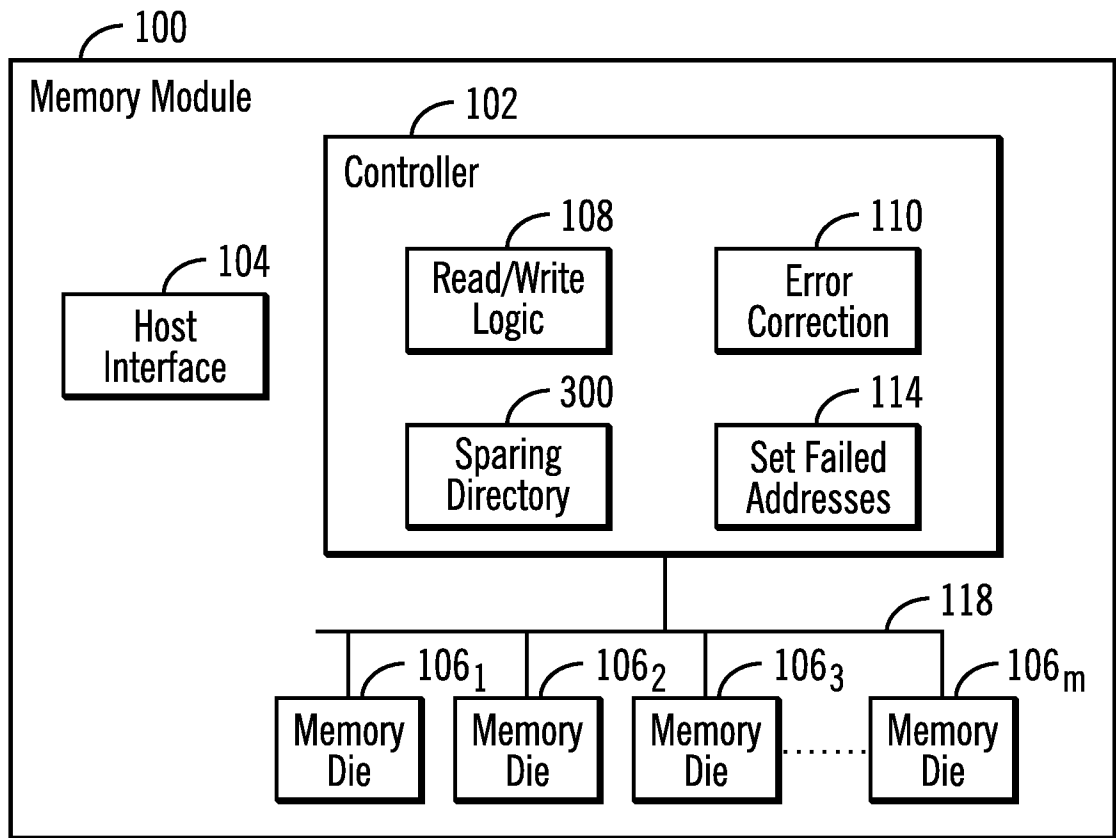


FIG. 1

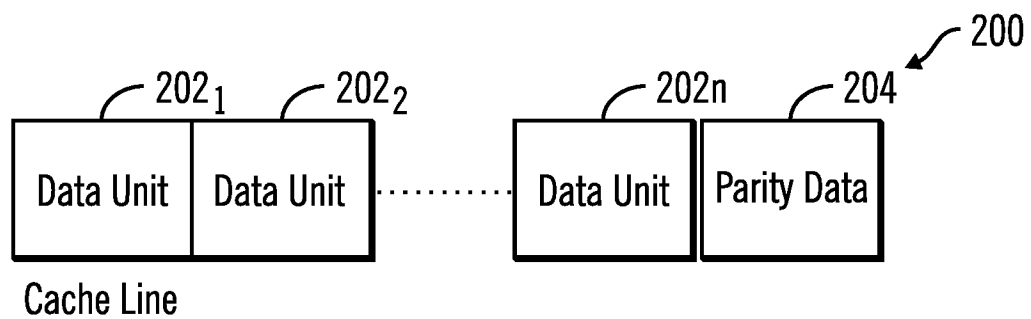


FIG. 2

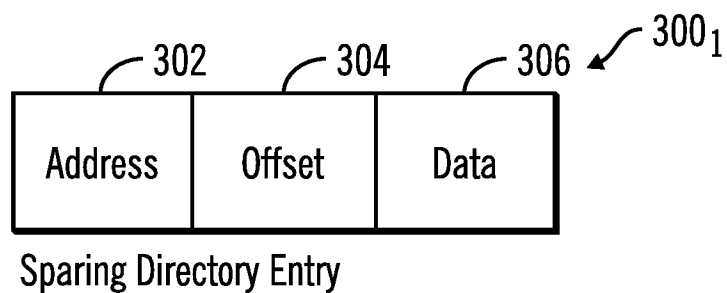


FIG. 3

2/5

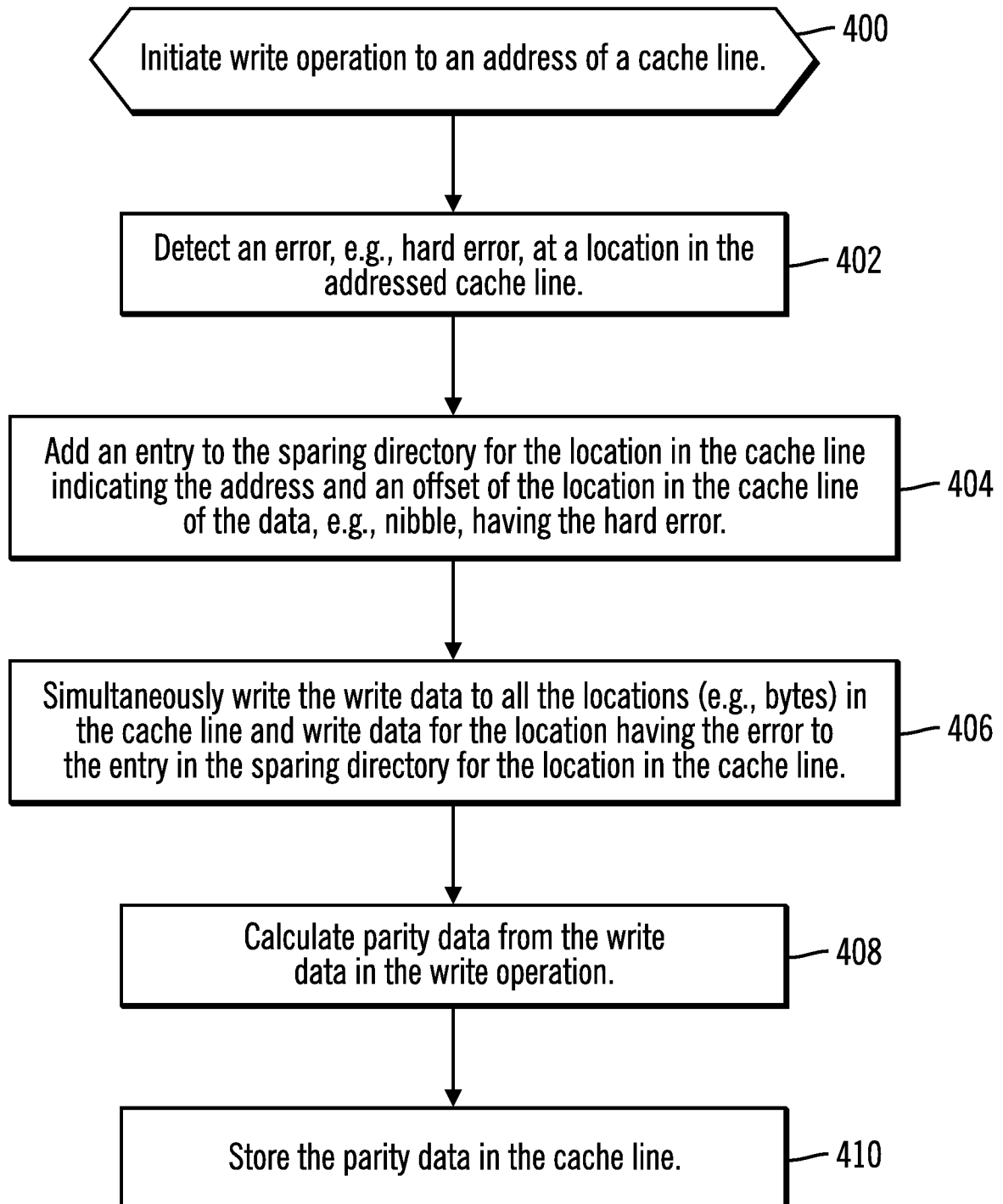


FIG. 4

3/5

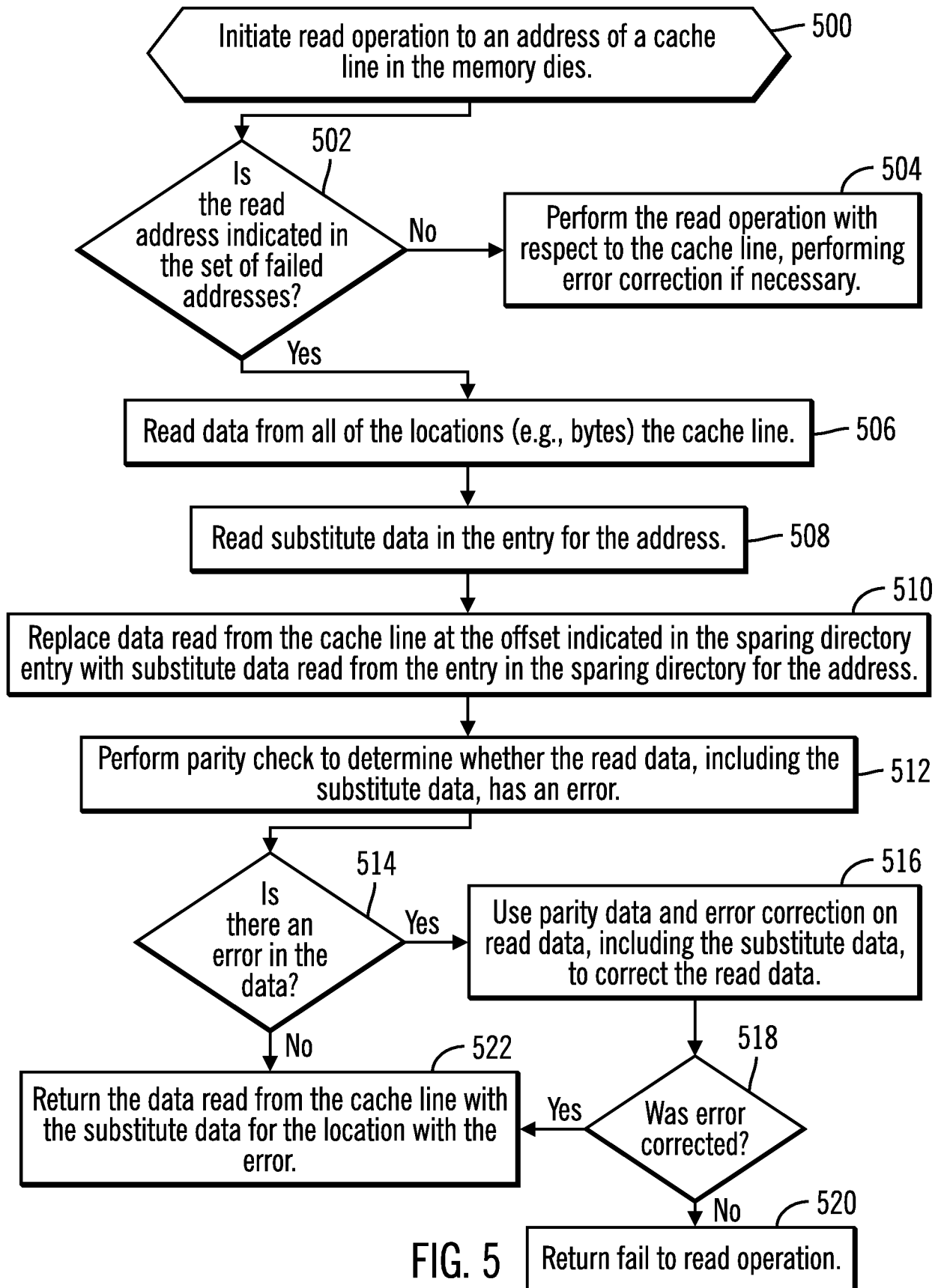


FIG. 5

4/5

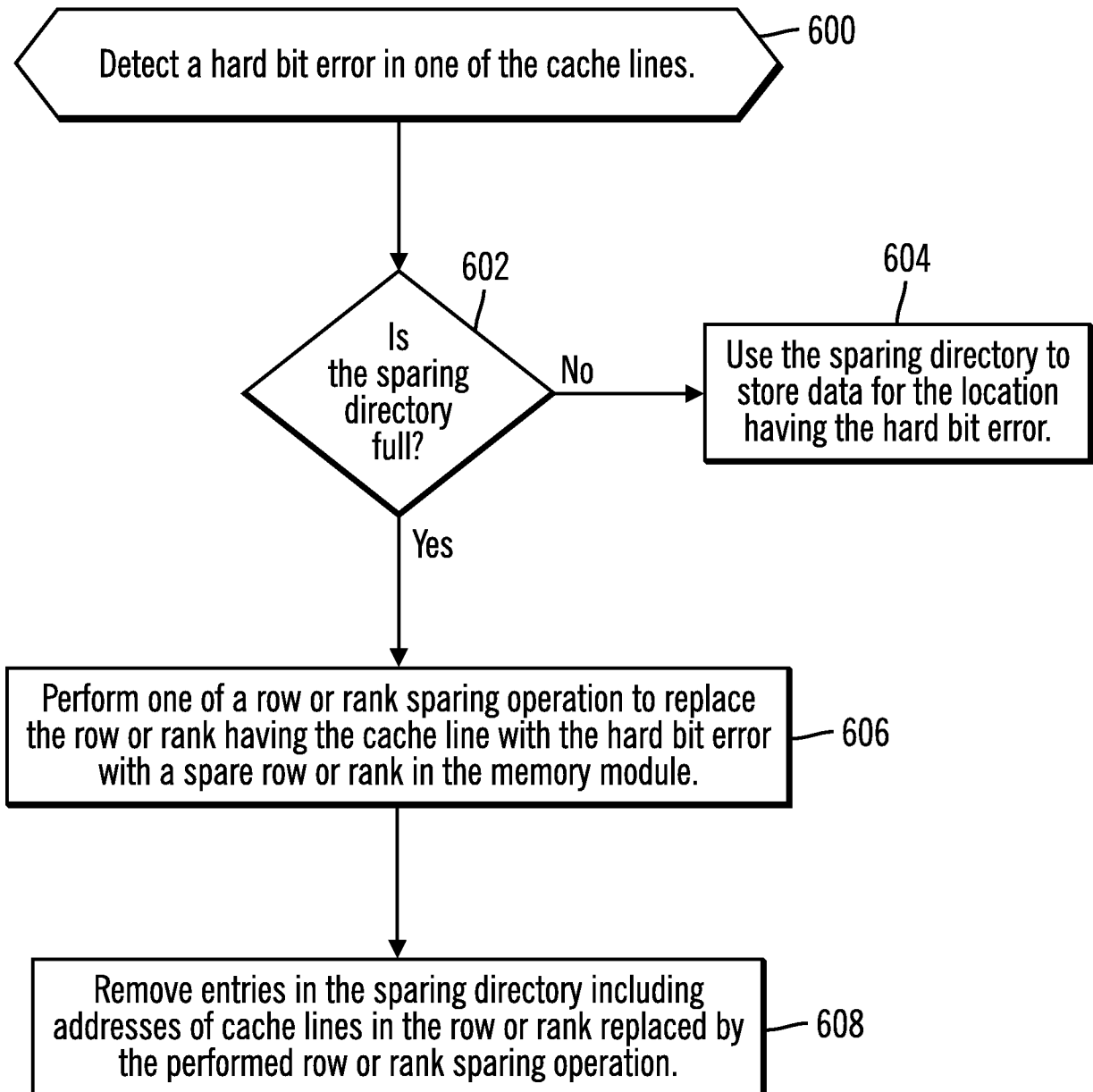


FIG. 6

5/5

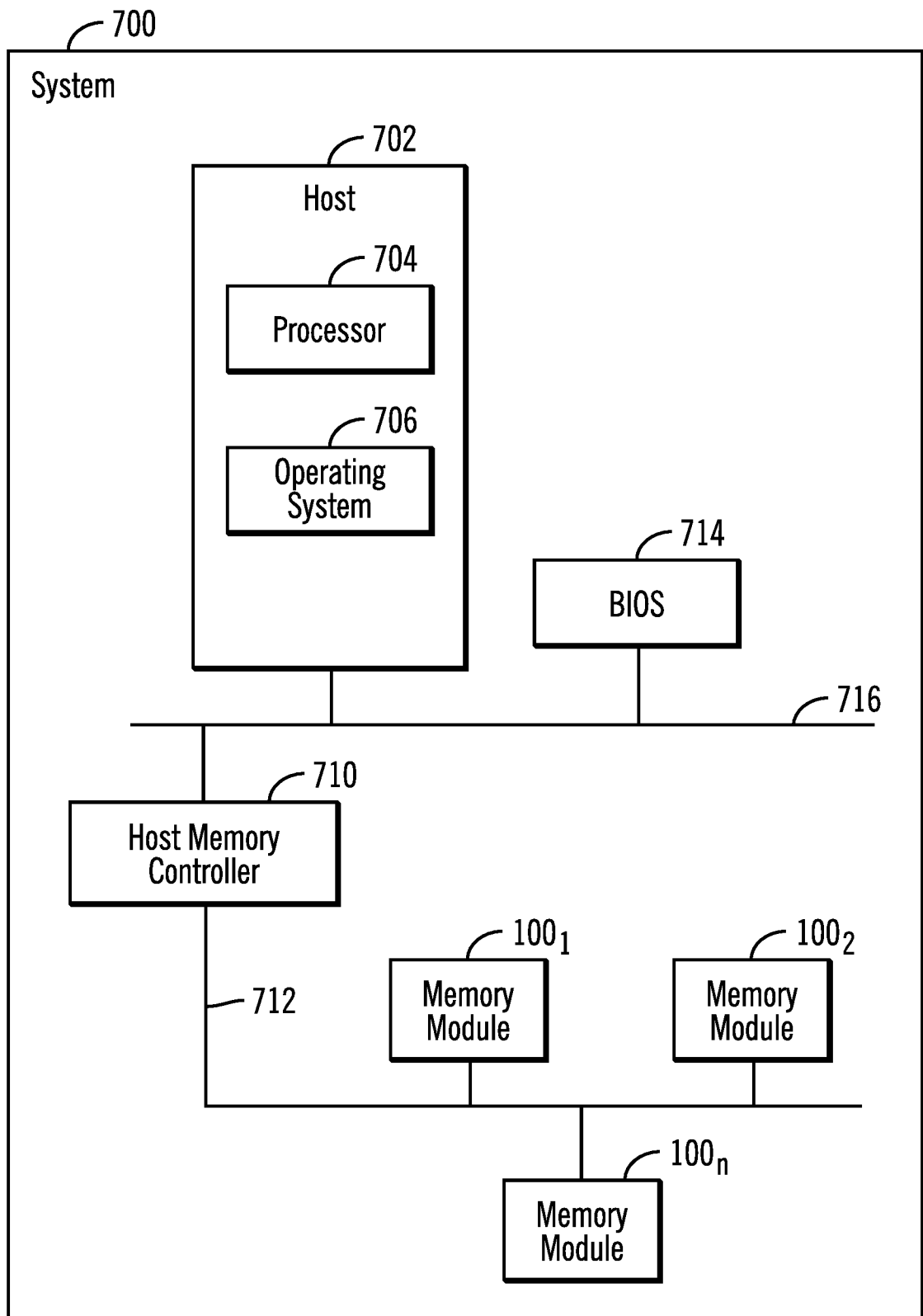


FIG. 7

A. CLASSIFICATION OF SUBJECT MATTER**G06F 11/07(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 11/07; G06F 12/02; G06F 12/00; G11C 29/00; G06F 11/16; G06F 11/20; G06F 11/00; G06F 12/08

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: memory, module, controller, cache line, write, read, sparing, error, entry, address, 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.
X	US 7,007,210 B2 (JAMES STEPHEN FIELDS, JR. et al.) 28 February 2006 See column 1, lines 45-47; column 2, lines 46-61; column 3, lines 26-63; column 4, line 3 - column 5, line 24; claims 1 and 19; and figures 1-5.	1-7, 10, 13-16, 18-22 , 24-25
A		8-9, 11-12, 17, 23
A	US 2013-0254506 A1 (STUART ALLEN BERKE et al.) 26 September 2013 See paragraphs [0019] and [0026]-[0030]; and figure 2.	1-25
A	US 2011-0099419 A1 (GREGG S. LUCAS et al.) 28 April 2011 See paragraphs [0013]-[0015] and [0018]; and figures 1-2.	1-25
A	US 2012-0254656 A1 (JOHN D. SCHOCK) 04 October 2012 See paragraphs [0034]-[0044] and figure 3.	1-25
A	WO 2015-088476 A1 (HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.) 18 June 2015 See paragraphs [0055]-[0058] and figure 7.	1-25



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

31 March 2017 (31.03.2017)

Date of mailing of the international search report

31 March 2017 (31.03.2017)

Name and mailing address of the ISA/KR

International Application Division

Korean Intellectual Property Office

189 Cheongsa-ro, Seo-gu, Daejeon, 35208, Republic of Korea

Facsimile No. +82-42-481-8578

Authorized officer

NHO, Ji Myong

Telephone No. +82-42-481-8528



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2016/063425

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 7007210 B2	28/02/2006	US 2003-0145257 A1	31/07/2003
US 2013-0254506 A1	26/09/2013	CN 104335187 A	04/02/2015
		EP 2828756 A1	28/01/2015
		EP 2828756 A4	22/04/2015
		EP 2828756 B1	18/01/2017
		IN 8056DEN2014 A	01/05/2015
		US 8719493 B2	06/05/2014
		WO 2013-142512 A1	26/09/2013
US 2011-0099419 A1	28/04/2011	US 7954021 B2	31/05/2011
US 2012-0254656 A1	04/10/2012	US 8667325 B2	04/03/2014
WO 2015-088476 A1	18/06/2015	TW 201523624 A	16/06/2015
		TW I539467 B	21/06/2016
		US 2016-0274968 A1	22/09/2016