



(51) International Patent Classification:
G11C 29/00 (2006.01) *G11C 16/02* (2006.01)

(21) International Application Number:
PCT/US2014/044354

(22) International Filing Date:
26 June 2014 (26.06.2014)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: HEWLETT-PACKARD DEVELOPMENT
COMPANY, L.P. [US/US]; 11445 Compaq Center Drive
W., Houston, TX 77070 (US).

(72) Inventors: WARNES, Lidia; 8000 Foothills Blvd., Ro-
seville, CA 95747 (US). BENEDICT, Melvin, K.; 11445
Compaq Center Dr W, Houston, TX 77070 (US).
WALTON, Andrew, C.; 8000 Foothills Blvd., Roseville,
CA 95747 (US).

(74) Agents: HA, Miranda, J. et al.; Hewlett-Packard Com-
pany, Intellectual Property Adminsitration, 3404 E. Har-
mony Road, Mail Stop 35, Fort Collins, CO 80528 (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,

[Continued on next page]

(54) Title: POST PACKAGE REPAIR (PPR) DATA IN NON-VOLATILE MEMORY

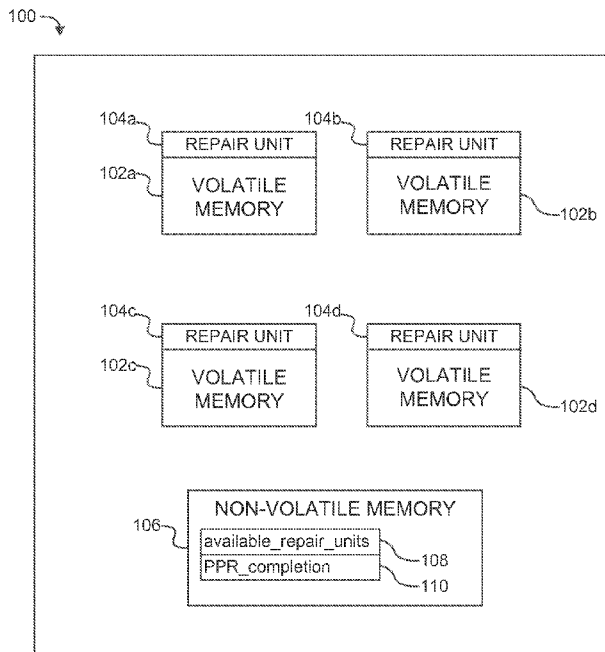


FIG. 1

(57) Abstract: Example implementations relate to post pack-
age repair (PPR) data in non- volatile memory. In example
implementations, PPR data may be stored in non-volatile
memory on a memory module. PPR data may indicate how
many of the plurality of repair units are available for per-
forming post package repair (PPR) on the plurality of volat-
ile memory devices, and how many PPRs have been per-
formed on the memory module.



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).

— *as to applicant's entitlement to apply for and be granted
a patent (Rule 4.17(ii))*

Declarations under Rule 4.17:

— *as to the identity of the inventor (Rule 4.17(i))*

Published:

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

- 1 -

POST PACKAGE REPAIR (PPR) DATA IN NON-VOLATILE MEMORY

BACKGROUND

[0001] A memory module may have multiple memory devices. Error correction logic may be used to correct errors detected in memory devices. A memory module may include extra memory cells so that when memory cells of a memory device on the memory module malfunction, extra memory cells may be used instead of the entire memory module having to be replaced.

BRIEF DESCRIPTION OF THE DRAWINGS

[0002] The following detailed description references the drawings, wherein:

[0003] FIG. 1 is a block diagram of an example memory module that includes a non-volatile memory having post package repair (PPR) data;

[0004] FIG. 2 is a block diagram of an example memory module that includes a plurality of available repair units fields for tracking availability of different types of repair units;

[0005] FIG. 3 is a block diagram of an example memory module that includes a plurality of available repair units fields for tracking availability of repair units on various volatile memory devices;

[0006] FIG. 4 is a block diagram of an example device that includes a machine-readable storage medium encoded with instructions to track PPRs performed on a memory module;

[0007] FIG. 5 is a block diagram of an example device that includes a machine-readable storage medium encoded with instructions to enable usage of a counter to track repair unit availability;

[0008] FIG. 6 is a block diagram of an example device that includes a machine-readable storage medium encoded with instructions to determine, based on a counter value, whether to initiate a PPR;

[0009] FIG. 7 is a flowchart of an example method for tracking PPRs; and

[0010] FIG. 8 is a flowchart of an example method for maintaining a data structure for PPR data.

DETAILED DESCRIPTION

[0011] Memory cells in memory devices may fail to properly store data, for example as a result of a manufacturing defect or due to environmental conditions as memory devices are used. A manufacturer of memory modules may fabricate redundant memory cells that may be used to replace defective memory cells in memory devices. Such redundant memory cells may be referred to herein as “repair units”. Repair units may include, for example, spare rows, spare columns, and/or spare single cells. The process of replacing a defective memory cell or cells with a repair unit after a memory device is packaged may be referred to herein as “post package repair”, or “PPR”.

[0012] Each volatile memory device on a memory module may track availability of its own repair units. A memory controller that is unaware of how many repair units are available to a volatile memory device may not know if a PPR is possible on the volatile memory device until the memory controller sends a PPR command and receives either a notification that the PPR is complete, or that an attempt to perform the PPR has failed. A failed attempt to perform a PPR may waste time and resources of a memory controller as well as of a memory module.

[0013] In light of the above, the present disclosure provides for storing PPR data in a non-volatile memory on a memory module. As used herein, the term “PPR data” refers to data related to the ability to perform PPRs, or to the effects of PPRs. PPR data may include, for example, how many repair units are available on a memory module and how many PPRs have been performed on a memory module. The storing of PPR data in a non-volatile memory on a memory module may allow a memory controller to use resources more efficiently when initiating PPRs, and allow PPR data to be tracked at a system level rather than at an individual volatile memory device level. In addition, multiple PPRs performed on the same volatile memory device may be tracked, and persistent PPR data in a non-volatile memory may allow PPRs and repair unit availability to be tracked across reboots of a memory module without expending system resources to discover/recreate such information during each boot time.

[0014] Referring now to the figures, FIG. 1 is a block diagram of an example memory module 100 that includes a non-volatile memory having PPR data. Memory module 100 may be an in-line memory module, such as a single in-line

- 3 -

memory module (SIMM) or a dual in-line memory module (DIMM), or any memory module suitable for mounting memory integrated circuits (ICs). In FIG. 1, memory module 100 includes volatile memory devices 102a, 102b, 102c, and 102d; repair units 104a, 104b, 104c, and 104d; and non-volatile memory 106. As used herein, the terms "include", "have", and "comprise" are interchangeable and should be understood to have the same meaning.

[0015] Volatile memory devices 102a-d may be on ICs on memory module 100. In some implementations, volatile memory devices 102a-d may be dynamic random-access memory (DRAM) devices. Although four volatile memory devices are shown in FIG. 1, it should be understood that memory module 100 may include more volatile memory devices or less volatile memory devices, and that the concepts discussed herein may be applicable to a memory module having any number of volatile memory devices.

[0016] Repair units 104a-d on memory module 100 may be used to replace defective portions of volatile memory devices 102a-d during PPRs. In some implementations, repair unit 104a may be used to replace a memory cell or cells in volatile memory device 102a. Similarly, repair units 104b, 104c, and 104d may be used to replace memory cells in volatile memory devices 102b, 102c, and 102d, respectively. In some implementations, any of repair units 104a-d may be used to replace memory cells in any of volatile memory devices 102a-d. In some implementations, repair units 104a, 104b, 104c, and 104d may be embedded in volatile memory devices 102a, 102b, 102c, and 102d, respectively. Although four repair units are shown in FIG. 1, it should be understood that memory module 100 may include more repair units or less repair units, and that the concepts discussed herein may be applicable to a memory module having any number of repair units.

[0017] In some implementations, repair units 104a-d may include repair units of a first type and repair units of a second type. For example, repair units 104a and 104b may be spare rows, and repair units 104c and 104d may be spare columns or spare single cells. In some implementations, a volatile memory device on memory module 100 may have multiple repair units, either of the same type or of different types. For example, volatile memory 102a may have five spare rows, or five repair units made up of a combination of spare rows, spare columns, and spare single cells.

- 4 -

[0018] Non-volatile memory 106 may store data about volatile memory devices 102a-d and any other volatile memory devices on memory module 100. In some implementations, non-volatile memory 106 may include a serial presence detect (SPD) electrically erasable programmable read-only memory (EEPROM). An SPD EEPROM on a memory module may store SPD data, such as memory timings and/or identifications of memory types on the memory module. Non-volatile memory 106 may include available repair units field 108 and PPR completion field 110. In some implementations, a non-volatile memory on a memory module may include a plurality of available repair units fields and a plurality of PPR completion fields, as discussed further below with respect to FIG. 2.

[0019] Available repair units field 108 may indicate how many of the repair units (e.g., repair units 104a-d) on memory module 100 are available for performing PPR on volatile memory devices (e.g., volatile memory devices 102a-d) on memory module 100. A repair unit may be available for performing PPR if the repair unit has not already been used (e.g., in a previous PPR) to replace a memory cell or cells in a volatile memory device. For example, if memory module 100 has four repair units (e.g., repair units 104a-d) and none of them have been used in a PPR, available repair units field 108 may have a value of 4. After one of the four repair units (e.g., repair unit 104b) has been used in a PPR, available repair units field 108 may have a value of 3.

[0020] PPR completion field 110 may indicate how many PPRs have been performed on volatile memory devices (e.g., volatile memory devices 102a-d) on memory module 100. For example, if memory module 100 has four volatile memory devices 102a-d and no PPR has been performed on any of them, PPR completion field 110 may have a value of 0. After a first PPR has been performed on volatile memory device 102b and a second PPR has been performed on volatile memory device 102d, PPR completion field 110 may have a value of 2.

[0021] Available repair units field 108 and PPR completion field 110 may be read by a memory controller during boot time of memory module 100. In some implementations, available repair units field 108 and PPR completion field 110 may be in an SPD EEPROM on memory module 100, and a memory controller may read these fields along with SPD data (e.g., timing information) in the SPD EEPROM during boot time of memory module 100. For example, the memory

- 5 -

controller may read available repair units field 108 to determine whether a PPR can be performed during boot time of memory module 100. In some implementations, the memory controller may use data in available repair units field 108 and PPR completion field 110 to initialize counters on the memory controller, as discussed below with respect to FIG. 5. In some implementations, a Basic Input/Output System (BIOS), or other firmware stored as machine-readable instructions, may read and/or update available repair units field 108 and PPR completion field 110 in an SPD EEPROM, and use values in such fields to determine whether a PPR should be performed, as discussed below with respect to FIG. 6.

[0022] In some implementations, available repair units field 108 and PPR completion field 110 may be updated by a memory controller when memory module 100 is rebooted or in response to a command to power down memory module 100. For example, the memory controller may maintain counters during runtime of memory module 100 to keep track of how many PPRs are performed and how many repair units are used on memory module 100. When a command is issued to reboot or power down memory module 100, the memory controller may write a counter value, indicating how many repair units are still available on memory module 100, to available repair units field 108. The memory controller may write a counter value, indicating how many PPRs have been performed on memory module 100, to PPR completion field 110. Thus, non-volatile memory 106 on memory module 100 may have up-to-date information about repair unit availability and number of PPRs performed at the beginning of the next runtime of memory module 100.

[0023] FIG. 2 is a block diagram of an example memory module 200 that includes a plurality of available repair units fields for tracking availability of different types of repair units. Memory module 200 may be an in-line memory module, such as a SIMM or DIMM, or any memory module suitable for mounting memory ICs. In FIG. 2, memory module 200 includes volatile memory devices 202a, 202b, 202c, and 202d; repair units 204a, 204b, 204c, and 204d; and non-volatile memory 206. Volatile memory devices 202a-d in FIG. 2 may be analogous to (e.g., have functions and/or components similar to) volatile memory devices 102a-d, respectively, in FIG. 1.

- 6 -

[0024] Repair units 204a-d may include repair units of a first type and repair units of a second type. For example, repair units 204a and 204c may be spare columns, and repair units 204b and 204d may be spare rows. Memory module 200 may include additional repair units of the first and/or second types, and may also include other types of repair units (e.g., spare single cells).

[0025] Non-volatile memory 206 may include available repair units fields 208a and 208b, and PPR completion fields 210a and 210b. Available repair units fields 208a and 208b may indicate how many repair units of a first type and of a second type, respectively, are available for performing PPR on volatile memory devices (e.g., volatile memory devices 202a-d) on memory module 200. For example, memory module 200 may have four repair units 204a-d, with repair units 204a and 204c being spare columns, and repair units 204b and 204d being spare rows. Available repair units field 208a may indicate how many spare columns are available for performing PPR on volatile memory devices on memory module 200, and available repair units field 208b may indicate how many spare rows are available for performing PPR on volatile memory devices on memory module 200. If none of repair units 204a-d have been used in a PPR, available repair units field 208a may have a value of 2, and available repair units field 208b may have a value of 2. After one of the spare rows (e.g., repair unit 204b) has been used in a PPR, available repair units field 208a may have a value of 2, and available repair units field 208b may have a value of 1.

[0026] PPR completion fields 210a and 210b may indicate how many repair units of the first type and of the second type, respectively, have been used to perform PPRs on volatile memory devices (e.g., volatile memory devices 202a-d) on memory module 200. For example, PPR completion field 210a may indicate how many spare columns have been used to perform PPRs on memory module 200, and PPR completion field 210b may indicate how many spare rows have been used to perform PPRs on memory module 200. If memory module 200 has four volatile memory devices 202a-d and no PPR has been performed on any of them, PPR completion field 210a may have a value of 0, and PPR completion field 210b may also have a value of 0. After one of the spare rows (e.g., repair unit 204b) has been used in a PPR, PPR completion field 210a may have a value of 0, and PPR completion field 210b may have a value of 1.

- 7 -

[0027] In some implementations, available repair units fields 208a-b and PPR completion fields 210a-b may be updated by a memory controller when memory module 200 is rebooted or in response to a command to power down memory module 200, as discussed above with respect to FIG. 1. If a BIOS on a memory controller identifies a defective portion of a volatile memory device during boot time of memory module 200, a PPR may be performed on the volatile memory device using an appropriate repair unit (e.g., a spare column may be used to replace a defective column), and the BIOS may update an available repair unit field and a PPR completion field in non-volatile memory 206 accordingly. In some implementations, available repair units fields 208a-b and PPR completion fields 210a-b may be updated by a memory controller during runtime of memory module 200. For example, if a PPR that uses a spare row is initiated during runtime of memory module 200, the memory controller may update available repair unit field 208b and PPR completion field 210b later on during runtime, after the PPR has been completed.

[0028] In some implementations, memory module 200 may include status module 212. A module may include a set of instructions encoded on a machine-readable storage medium and executable by a processor of memory module 200. In addition or as an alternative, a module may include a hardware device comprising electronic circuitry for implementing the functionality described below.

[0029] Status module 212 may transmit, in response to a PPR query directed at one of volatile memory devices 202a-d, a status indication for the volatile memory device at which the PPR query is directed. As used herein, the term "PPR query" should be understood to refer to a query regarding a volatile memory device's ability to perform a PPR. The volatile memory device at which the PPR query is directed may be referred to herein as the "targeted volatile memory device". Status module 212 may transmit an indication of PPR availability, if there is no PPR being performed on the targeted volatile memory device, and a repair unit on memory module 200 is available for performing PPR on the targeted volatile memory device, when the PPR query is received by memory module 200. Status module 212 may transmit an in-progress indication, if a PPR is being performed on the targeted volatile memory device when the PPR query is received by memory module 200. Status module 212 may transmit an indication that no PPR is

- 8 -

available, if none of the repair units on memory module 200 is available for performing PPR on the targeted volatile memory device when the PPR query is received by memory module 200.

[0030] FIG. 3 is a block diagram of an example memory module 300 that includes a plurality of available repair units fields for tracking availability of repair units on various volatile memory devices. Memory module 300 may be an in-line memory module, such as a SIMM or DIMM, or any memory module suitable for mounting memory ICs. In FIG. 3, memory module 300 includes volatile memory devices 1, 2, 3, and 4, each of which may have a plurality of repair units (304a, 304b, 304c, and 304d, respectively) used to perform PPRs on the respective volatile memory device. Each of volatile memory devices 1-4 in FIG. 3 may be analogous to any of the volatile memory devices discussed above with respect to FIGS. 1 and 2. Each of the repair units in FIG. 3 may be analogous to any of the repair units discussed above with respect to FIGS. 1 and 2.

[0031] In FIG. 3, memory module 300 includes non-volatile memory 306, which may include available repair units fields 308a, 308b, 308c, and 308d. Available repair units fields 308a, 308b, 308c, and 308d may correspond to volatile memory devices 1, 2, 3, and 4, respectively. Each of available repair units fields 308a-d may indicate how many of a plurality of repair units on memory module 300 are available for performing PPR on the respective volatile memory device. For example, each of the plurality of repair units 304a-d may have five repair units. If none of the repair units on memory module 300 have been used to perform a PPR, each of available repair units fields 308a-d may have a value of 5. After a PPR is performed on volatile memory device 2 and three PPRs are performed on volatile memory device 3, available repair units fields 308a, 308b, 308c, and 308d may have a value of 5, 4, 2, and 5, respectively. Available repair units fields 308a-d may be read by a memory controller during boot time of memory module 300 to determine how many PPRs can be performed on each volatile memory device on memory module 300.

[0032] FIG. 4 is a block diagram of an example device 400 that includes a machine-readable storage medium encoded with instructions to track PPRs performed on a memory module. Device 400 may be a memory controller. In some implementations, device 400 may be part of an electronic user device, such as a

notebook computer, a desktop computer, a workstation, a tablet computing device, a mobile phone, or an electronic book reader. In some implementations, device 400 may be part of a server. In FIG. 4, device 400 includes processor 402 and machine-readable storage medium 404.

[0033] Processor 402 may include a central processing unit (CPU), microprocessor (e.g., semiconductor-based microprocessor), and/or other hardware device suitable for retrieval and/or execution of instructions stored in machine-readable storage medium 404. Processor 402 may fetch, decode, and/or execute instructions 406, 408, and 410. As an alternative or in addition to retrieving and/or executing instructions, processor 402 may include an electronic circuit comprising a number of electronic components for performing the functionality of instructions 406, 408, and/or 410.

[0034] Machine-readable storage medium 404 may be any suitable electronic, magnetic, optical, or other physical storage device that contains or stores executable instructions. Thus, machine-readable storage medium 404 may include, for example, a random-access memory (RAM), an EEPROM, a storage device, an optical disc, and the like. In some implementations, machine-readable storage medium 404 may include a non-transitory storage medium, where the term "non-transitory" does not encompass transitory propagating signals. As described in detail below, machine-readable storage medium 404 may be encoded with a set of executable instructions 406, 408, and 410.

[0035] Instructions 406 may read PPR data, for a memory module, from non-volatile memory on the memory module. The PPR data for the memory module may include an indication of how many PPRs have been performed on the memory module. In some implementations, the PPR data may also include an indication of how many repair units are available for performing PPR on the memory module. In some implementations, PPR data for a memory module may include a manufacturer-provided indication, in non-volatile memory of the memory module, of how many repair units were fabricated on the memory module. Execution of instructions 406 may cause, for example, contents of non-volatile memory 106, 206, or 306 to be read. In some implementations, instructions 406 may be executed during boot time of a memory module. In some implementations, instructions 406 may be executed during runtime of a memory module.

- 10 -

[0036] Instructions 408 may determine, based on PPR data read from non-volatile memory on a memory module, how many repair units are available for performing PPR on the memory module. In some implementations, instructions 408 may identify indications, in the PPR data, of how many repair units are available. For example, instructions 406 may read available repair units field 108, and instructions 408 may identify the value in available repair units field 108. In some implementations, PPR data may be read that includes an indication of how many repair units the memory module has (e.g., how many repair units the memory module is manufactured with), and an indication of how many PPRs have been performed on the memory module. In such implementations, instructions 408 may determine how many repair units are available for performing PPR by subtracting the number of PPRs performed from the number of repair units the memory module has.

[0037] Instructions 410 may track PPRs performed on a memory module. For example, instructions 410 may increment a counter every time a repair unit is used to perform a PPR on the memory module. The counter may be incremented when a PPR completion indication is received from the memory module. In some implementations, instructions 410 may track PPRs performed on individual volatile memory devices on a memory module, maintaining a different counter for each volatile memory device. In some implementations, instructions 410 may track PPRs performed on multiple memory modules, maintaining a different counter for each memory module.

[0038] FIG. 5 is a block diagram of an example device 500 that includes a machine-readable storage medium encoded with instructions to enable usage of a counter to track repair unit availability. Device 500 may be a memory controller. In some implementations, device 500 may be part of an electronic user device, such as a notebook computer, a desktop computer, a workstation, a tablet computing device, a mobile phone, or an electronic book reader. In some implementations, device 500 may be part of a server. In FIG. 5, device 500 includes processor 502 and machine-readable storage medium 504.

[0039] As with processor 402 of FIG. 4, processor 502 may include a CPU, microprocessor (e.g., semiconductor-based microprocessor), and/or other hardware device suitable for retrieval and/or execution of instructions stored in machine-readable storage medium 504. Processor 502 may fetch, decode, and/or

execute instructions 506, 508, 510, 512, and 514 to enable usage of a counter to track repair unit availability, as described below. As an alternative or in addition to retrieving and/or executing instructions, processor 502 may include an electronic circuit comprising a number of electronic components for performing the functionality of instructions 506, 508, 510, 512, and/or 514.

[0040] As with machine-readable storage medium 404 of FIG. 4, machine-readable storage medium 504 may be any suitable physical storage device that stores executable instructions. Instructions 506, 508, and 510 on machine-readable storage medium 504 may be analogous to instructions 406, 408, and 410 on machine-readable storage medium 404. Instructions 512 may initialize, based on PPR data read from non-volatile memory on a memory module, a plurality of counters. The plurality of counters may be initialized during boot time of the memory module. In some implementations, the plurality of counters may be stored in a data structure on a memory controller. The plurality of counters may include an available repair units counter and a PPR completion counter. The available repair units counter may track how many repair units are available for performing PPR on the memory module. The PPR completion counter may track how many PPRs have been performed on the memory module.

[0041] For example, execution of instructions 506 may cause available repair units field 108 and PPR completion field 110 to be read from non-volatile memory 106 of memory module 100. Instructions 512 may initialize an available repair units counter to have a value equal to the value of available repair units field 108. In some implementations, an available repair units counter for a memory module may be initialized with a value equal to the number of repair units manufactured on the memory module. Instructions 512 may initialize a PPR completion counter to have a value equal to the value of PPR completion field 110. In some implementations, a memory module from which PPR data is read may have repair units of different types. In such implementations, an available repair units field and a PPR completion field (e.g., available repair units fields 208a-b and PPR completion fields 210a-b of FIG. 2) may be read for each repair unit type, and instructions 512 may use the field values to initialize an available repair units counter and a PPR completion counter for each repair unit type. In some implementations, available repair units fields and PPR completion fields corresponding to various volatile memory devices on a memory

module may be read from non-volatile memory on the memory module, and instructions 512 may use the field values to initialize an available repair units counter and a PPR completion counter for each volatile memory device.

[0042] Instructions 514 may update a plurality of counters as PPRs are performed on a memory module. For example, instructions 514 may include instructions to decrement an available repair units counter, and increment a PPR completion counter, in response to receiving a PPR completion indication or an in-progress indication. The PPR completion indication or in-progress indication may be received in response to a PPR query issued by a memory controller to a volatile memory device on a memory module. If different counters are maintained for different volatile memory devices and/or repair unit types, instructions 514 may increment/decrement the appropriate counters based on information in the PPR completion/in-progress indication, and/or information in the PPR query in response to which the PPR completion/in-progress indication was received, that identifies the targeted volatile memory device and/or the repair unit type used in the PPR corresponding to the PPR completion/in-progress indication.

[0043] In some implementations, a PPR may be performed on a volatile memory device on a memory module without a command to perform a PPR being issued by a memory controller. Since the memory controller did not issue a command for the PPR that is performed, the memory controller may not receive a PPR completion indication when the PPR is completed, and an available repair units counter maintained by the memory controller may not be decremented after the PPR is completed, even though there is one less repair unit available after the PPR is completed. A situation may arise where there are no more repair units available for PPR even though an available repair units counter maintained by the memory controller has a non-zero value. Thus, instructions 514 may include instructions to set an available repair units counter to zero in response to receiving an indication that no PPR is available.

[0044] FIG. 6 is a block diagram of an example device 600 that includes a machine-readable storage medium encoded with instructions to determine, based on a counter value, whether to initiate a PPR. In some implementations, device 600 may be part of an electronic user device, such as a notebook computer, a desktop computer, a workstation, a tablet computing device, a mobile phone, or an electronic

book reader. In some implementations, device 600 may be part of a server. In some implementations, device 600 may be implemented on a server. In FIG. 6, device 600 includes processor 602 and machine-readable storage medium 604.

[0045] As with processor 402 of FIG. 4, processor 602 may include a CPU, microprocessor (e.g., semiconductor-based microprocessor), and/or other hardware device suitable for retrieval and/or execution of instructions stored in machine-readable storage medium 604. Processor 602 may fetch, decode, and/or execute instructions 606, 608, 610, 612, and 614. As an alternative or in addition to retrieving and/or executing instructions, processor 602 may include an electronic circuit comprising a number of electronic components for performing the functionality of instructions 606, 608, 610, 612, and/or 614.

[0046] As with machine-readable storage medium 404 of FIG. 4, machine-readable storage medium 604 may be any suitable physical storage device that stores executable instructions. Instructions 606, 608, and 610 on machine-readable storage medium 604 may be analogous to instructions 406, 408, and 410 on machine-readable storage medium 404. Instructions 612 may determine, based on a value of one of a plurality of counters, whether to initiate a PPR on a memory module. For example, instructions 612 may identify a pattern of corrected memory errors or a number of repeated error interrupts as a situation for which a PPR is appropriate. In some implementations, instructions 612 may determine that a PPR is to be initiated if the value of an available repair units counter is greater than zero, and that a PPR is not to be initiated if the value of the available repair units counter is equal to zero. The available repair units counter may correspond to a memory module or a specific volatile memory device on which a defective memory cell or cells are identified. A PPR may be initiated during boot time or runtime of a memory module.

[0047] In some implementations, instructions 612 may determine that a PPR is to be initiated if the value of a PPR completion counter is less than the number of repair units on a memory module or a specific volatile memory device corresponding to the PPR completion counter. Instructions 612 may determine that a PPR is not to be initiated if the value of the PPR completion counter is equal to or greater than the number of repair units on the memory module or specific volatile memory device. The relevant memory module or specific volatile memory device may be one on which a defective memory cell or cells are identified. If it is determined that a PPR is not to be

initiated, a memory controller may use methods such as memory erasure or double device data correction (DDDC) to correct errors caused by the defective memory cell(s), or may request a service call.

[0048] Instructions 614 may write respective values of an available repair units counter and a PPR completion counter to a non-volatile memory on a memory module. In some implementations, the counters may be incremented/decremented each time a PPR is completed, and the new values of the counters may be written to the non-volatile memory after each PPR is completed. Counter values may be written to the non-volatile memory during boot time and/or runtime of the memory module. In some implementations, counters may be incremented/decremented each time a PPR is completed during runtime of the memory module, but the counter values might not be written to the non-volatile memory until a command to reboot or power down the memory module is issued. In some implementations, the counters may be stored in a data structure on a memory controller.

[0049] Methods related to storing PPR data in non-volatile memory are discussed with respect to FIGS. 7-8. FIG. 7 is a flowchart of an example method 700 for tracking PPRs. Although execution of method 700 is described below with reference to processor 502 of FIG. 5, it should be understood that execution of method 700 may be performed by other suitable devices, such as processors 402 and 602 of FIGS. 4 and 6, respectively. Method 700 may be implemented in the form of executable instructions stored on a machine-readable storage medium and/or in the form of electronic circuitry.

[0050] Method 700 may start in block 702, where processor 502 may read PPR data, for a first memory module, from non-volatile memory on the first memory module. The PPR data for the first memory module may include an indication of how many PPRs have been performed on the first memory module. For example, the PPR data for the first memory module may include a value of a PPR completion field in non-volatile memory on the first memory module. In some implementations, PPR data for the first memory module may be read, during boot time of the first memory module, from an SPD EEPROM on the first memory module.

[0051] In block 704, processor 502 may read PPR data, for a second memory module, from non-volatile memory on the second memory module. The PPR data for the second memory module may include an indication of how many PPRs have been

performed on the second memory module. For example, the PPR data for the second memory module may include a value of a PPR completion field in non-volatile memory on the second memory module. In some implementations, PPR data for the second memory module may be read, during boot time of the second memory module, from an SPD EEPROM on the second memory module. Although block 704 is shown below block 702 in FIG. 7, it should be understood that elements of block 704 may be performed before or in parallel with elements of block 702.

[0052] In block 706, processor 502 may maintain a data structure for tracking PPRs performed on the first and second memory modules. Fields of the data structure may be initialized using the indications, read from non-volatile memory on the first and second memory modules, of how many PPRs have been performed on the first and second memory modules, respectively. In some implementations, the data structure may include a first available repair units field, a second available repair units field, a third available repair units field, and a fourth available repair units field. The first and second available repair units fields may indicate how many repair units of a first type and of a second type, respectively, are available for performing PPR on the first memory module. The third and fourth available repair units fields may indicate how many repair units of the first type and of the second type, respectively, are available for performing PPR on the second memory module.

[0053] In some implementations, the data structure may include a first PPR completion field, a second PPR completion field, a third PPR completion field, and a fourth PPR completion field. The first and second PPR completion fields may indicate how many repair units of the first type and of the second type, respectively, have been used to perform PPRs on the first memory module. The third and fourth PPR completion fields may indicate how many repair units of the first type and of the second type, respectively, have been used to perform PPRs on the second memory module. In some implementations, fields of the data structure may correspond to counters maintained by a memory controller, as discussed above with respect to FIGS. 5-6.

[0054] FIG. 8 is a flowchart of an example method 800 for maintaining a data structure for PPR data. Although execution of method 800 is described below with reference to processor 602 of FIG. 6, it should be understood that execution of method 800 may be performed by other suitable devices, such as processors 402

- 16 -

and 502 of FIGS. 4 and 5, respectively. Some blocks of method 800 may be performed in parallel with and/or after method 700. Method 800 may be implemented in the form of executable instructions stored on a machine-readable storage medium and/or in the form of electronic circuitry.

[0055] Method 800 may start in block 802, where processor 602 may initialize fields of a data structure for tracking PPRs. Fields of the data structure may include available repair units fields and PPR completion fields, as discussed above with respect to FIG. 7. Fields of the data structure may be initialized in a manner analogous to that of initializing counters, as discussed above with respect to FIG. 5. Fields of the data structure may be initialized during boot time of a memory module.

[0056] In block 804, processor 602 may determine, based on values of fields in the data structure, whether to initiate PPRs on the first and second memory modules. In some implementations, processor 602 may determine that a PPR is to be initiated if the value of an available repair units field in the data structure is greater than zero, and that a PPR is not to be initiated if the value of the available repair units field is equal to zero. A first available repair units field may correspond to the first memory module, and a second available repair units field may correspond to the second memory module; a decision may be made regarding whether to initiate a PPR on the first or second memory module based on the value of the respective available repair units field.

[0057] In block 806, processor 602 may update fields in the data structure in response to status indications and PPR completion indications. Fields in the data structure may be updated in a manner analogous to that of updating counters, as discussed above with respect to FIG. 5. Although block 806 is shown below block 804 in FIG. 8, it should be understood that elements of block 806 may be performed before or in parallel with elements of block 804.

[0058] The foregoing disclosure describes storing PPR data in a non-volatile memory on a memory module. Example implementations described herein enable more efficient use of resources when initiating and tracking PPRs, and tracking of PPR data at a system level rather than at an individual volatile memory device level.

- 17 -

We claim:

1. A memory module comprising:
a plurality of volatile memory devices;
a plurality of repair units; and
non-volatile memory to store data about the plurality of volatile memory devices, wherein the non-volatile memory comprises:

an available repair units field to indicate how many of the plurality of repair units are available for performing post package repair (PPR) on the plurality of volatile memory devices; and

a PPR completion field to indicate how many PPRs have been performed on the plurality of volatile memory devices;

wherein the available repair units field and the PPR completion field are to be read by a memory controller during boot time of the memory module.

2. The memory module of claim 1, wherein the non-volatile memory comprises a serial presence detect (SPD) electrically erasable programmable read-only memory (EEPROM).

3. The memory module of claim 1, wherein:

the plurality of repair units comprise repair units of a first type and repair units of a second type;

the non-volatile memory comprises a plurality of available repair units fields and a plurality of PPR completion fields;

the plurality of available repair units fields comprise a first available repair units field and a second available repair units field to indicate how many repair units of the first type and of the second type, respectively, are available for performing PPR on the plurality of volatile memory devices; and

the plurality of PPR completion fields comprise a first PPR completion field and a second PPR completion field to indicate how many repair units of the first type and of the second type, respectively, have been used to perform PPRs on the plurality of volatile memory devices.

- 18 -

4. The memory module of claim 1, wherein the available repair units field and the PPR completion field are updated by the memory controller when the memory module is rebooted or in response to a command to power down the memory module.

5. The memory module of claim 1, further comprising a status module to transmit, in response to a PPR query directed at one of the plurality of volatile memory devices:

an indication of PPR availability, if there is no PPR being performed on the one of the plurality of volatile memory devices, and one of the plurality of repair units is available for performing PPR on the one of the plurality of volatile memory devices, when the PPR query is received by the memory module;

an in-progress indication, if a PPR is being performed on the one of the plurality of volatile memory devices when the PPR query is received by the memory module; and

an indication that no PPR is available, if none of the plurality of repair units is available for performing PPR on the one of the plurality of volatile memory devices when the PPR query is received by the memory module.

6. The memory module of claim 1, wherein the non-volatile memory comprises a plurality of available repair units fields, each of the plurality of available repair units fields corresponding to a respective one of the plurality of volatile memory devices and indicating how many of the plurality of repair units are available for performing PPR on the respective one of the plurality of volatile memory devices.

7. A machine-readable storage medium encoded with instructions executable by a processor, the machine-readable storage medium comprising:

instructions to read post package repair (PPR) data, for a memory module, from non-volatile memory on the memory module, wherein the PPR data for the memory module comprises an indication of how many PPRs have been performed on the memory module;

- 19 -

instructions to determine, based on PPR data read from non-volatile memory on the memory module, how many repair units are available for performing PPR on the memory module; and

instructions to track PPRs performed on the memory module.

8. The machine-readable storage medium of claim 7, wherein the PPR data further comprises an indication of how many repair units are available for performing PPR on the memory module.

9. The machine-readable storage medium of claim 7, further comprising:

instructions to initialize, based on PPR data read from non-volatile memory on the memory module, a plurality of counters, the plurality of counters comprising:

an available repair units counter to track how many repair units are available for performing PPR on the memory module; and

a PPR completion counter to track how many PPRs have been performed on the memory module; and

instructions to update the plurality of counters as PPRs are performed on the memory module.

10. The machine-readable storage medium of claim 9, wherein instructions to update the plurality of counters comprise:

instructions to decrement the available repair units counter, and increment the PPR completion counter, in response to receiving a PPR completion indication or an in-progress indication; and

instructions to set the available repair units counter to zero in response to receiving an indication that no PPR is available.

11. The machine-readable storage medium of claim 9, further comprising:

instructions to determine, based on a value of one of the plurality of counters, whether to initiate a PPR on the memory module; and

instructions to write respective values of the available repair units counter and the PPR completion counter to the non-volatile memory on the memory module.

- 20 -

12. A method comprising:

reading post package repair (PPR) data, for a first memory module, from non-volatile memory on the first memory module, wherein the PPR data for the first memory module comprises an indication of how many PPRs have been performed on the first memory module;

reading PPR data, for a second memory module, from non-volatile memory on the second memory module, wherein the PPR data for the second memory module comprises an indication of how many PPRs have been performed on the second memory module;

maintaining a data structure for tracking PPRs performed on the first and second memory modules, wherein fields of the data structure are initialized using the indications, read from non-volatile memory on the first and second memory modules, of how many PPRs have been performed on the first and second memory modules, respectively.

13. The method of claim 12, wherein:

PPR data for the first memory module is read, during boot time of the first memory module, from a serial presence detect (SPD) electrically erasable programmable read-only memory (EEPROM) on the first memory module; and

PPR data for the second memory module is read, during boot time of the second memory module, from an SPD EEPROM on the second memory module.

14. The method of claim 12, wherein the data structure for tracking PPRs comprises:

a first available repair units field and a second available repair units field to indicate how many repair units of a first type and of a second type, respectively, are available for performing PPR on the first memory module;

a third available repair units field and a fourth available repair units field to indicate how many repair units of the first type and of the second type, respectively, are available for performing PPR on the second memory module;

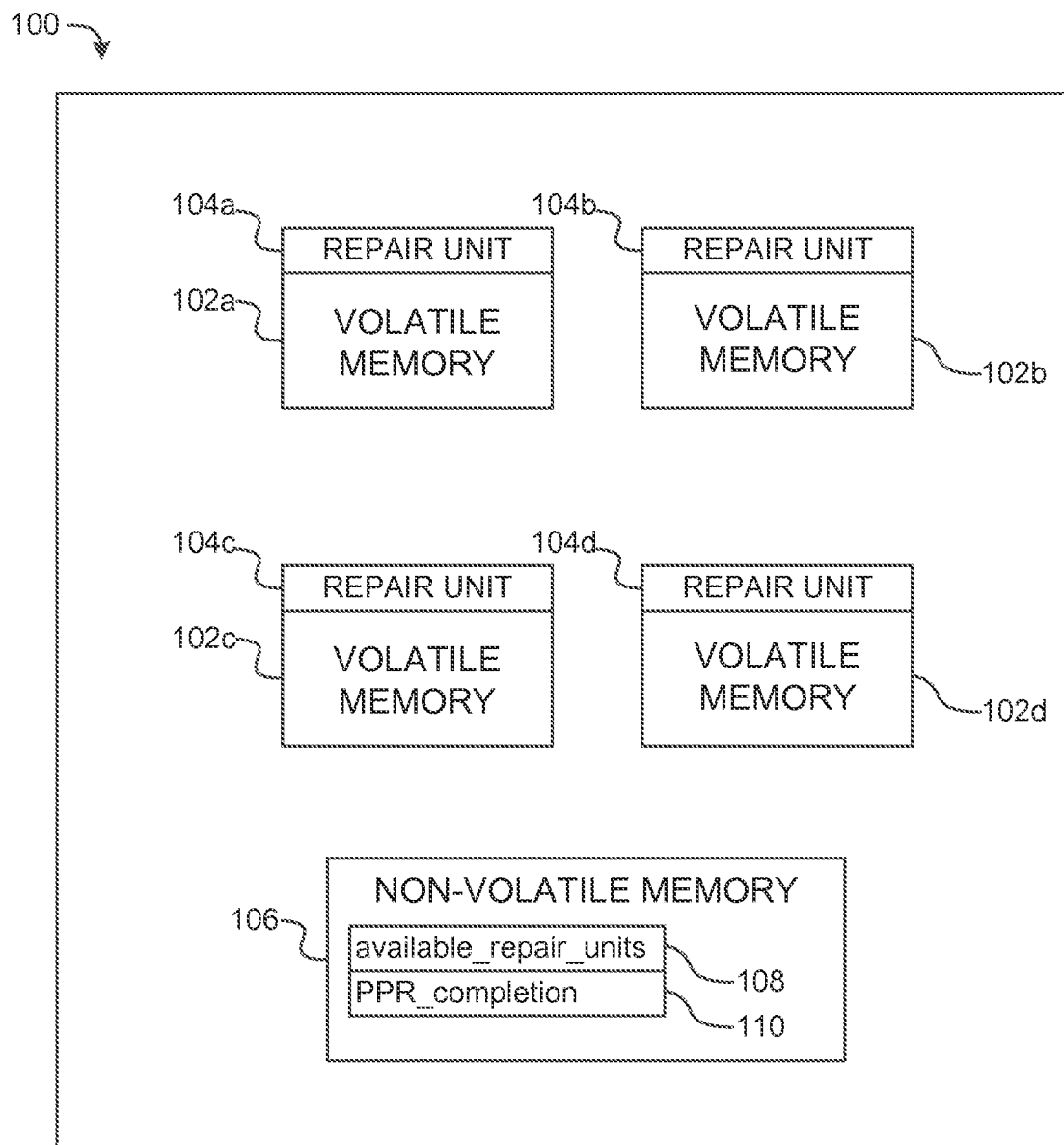
- 21 -

a first PPR completion field and a second PPR completion field to indicate how many repair units of the first type and of the second type, respectively, have been used to perform PPRs on the first memory module; and

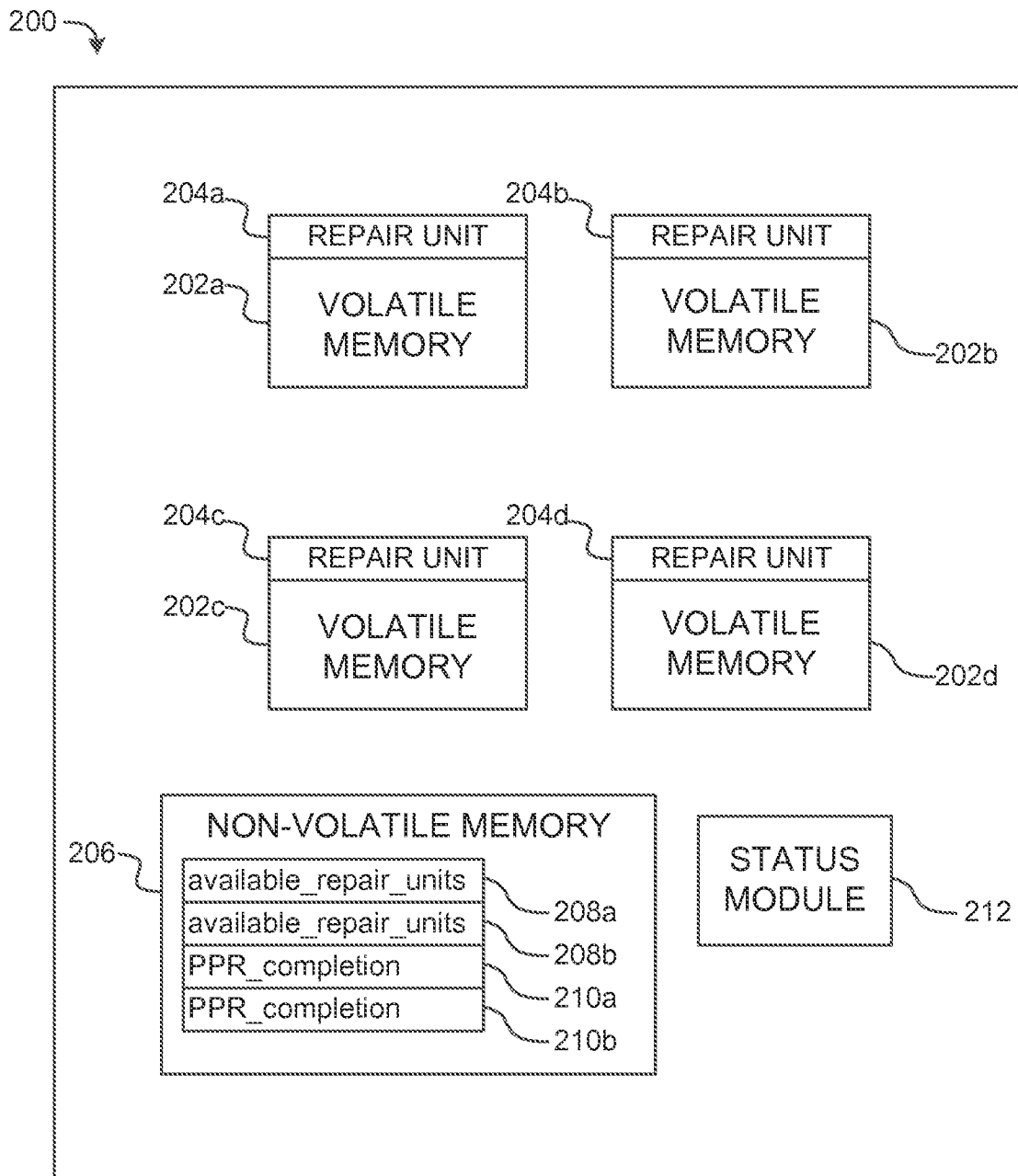
a third PPR completion field and a fourth PPR completion field to indicate how many repair units of the first type and of the second type, respectively, have been used to perform PPRs on the second memory module.

15. The method of claim 14, wherein maintaining the data structure comprises updating fields in the data structure in response to status indications and PPR completion indications, the method further comprising determining, based on values of fields in the data structure, whether to initiate PPRs on the first and second memory modules.

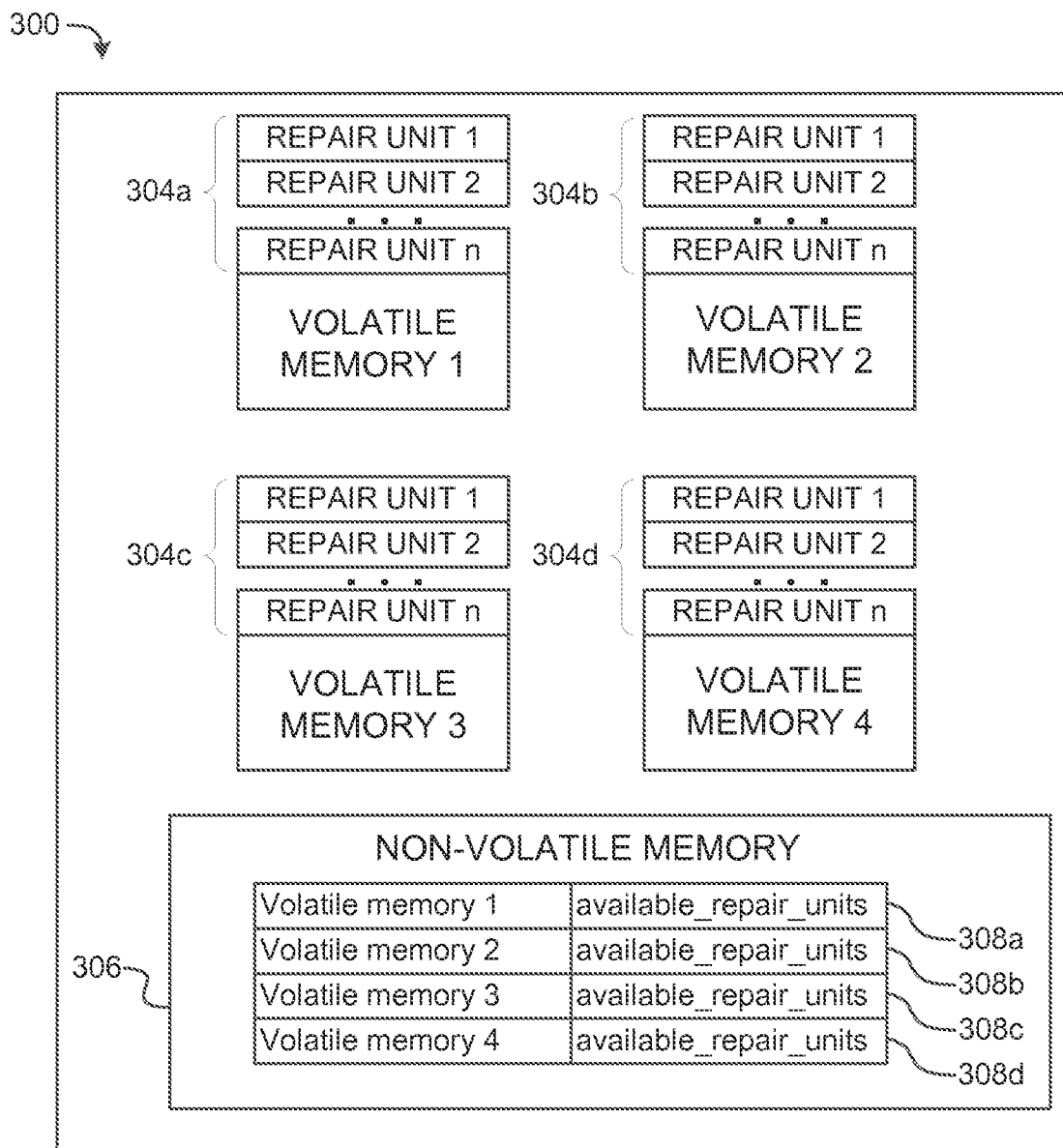
1/7

**FIG. 1**

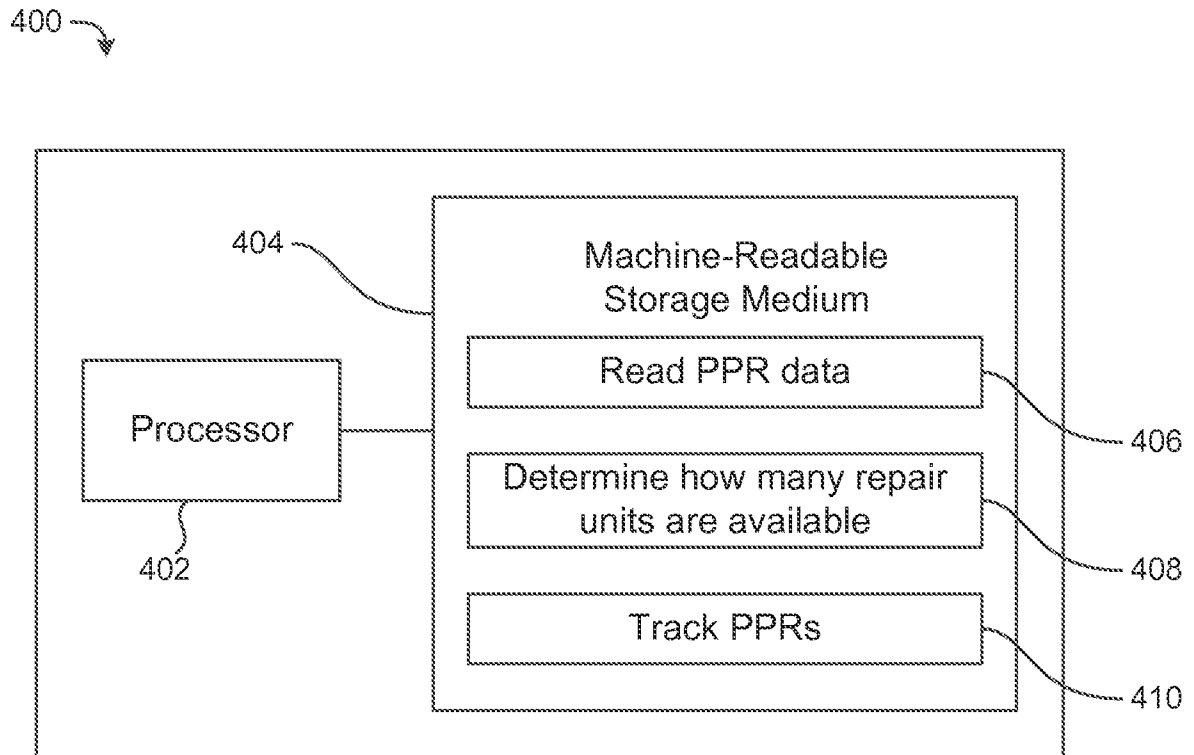
2/7

**FIG. 2**

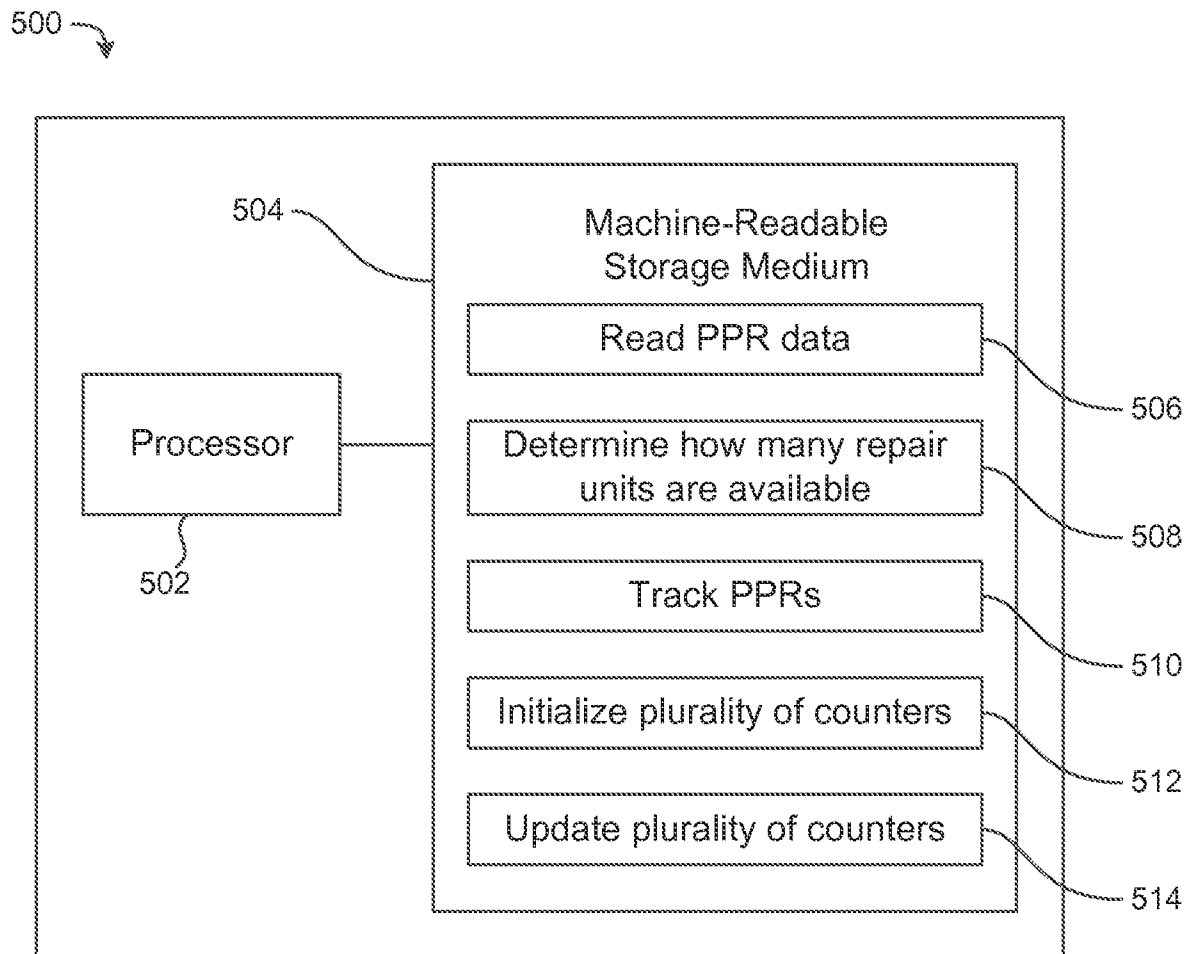
3/7

**FIG. 3**

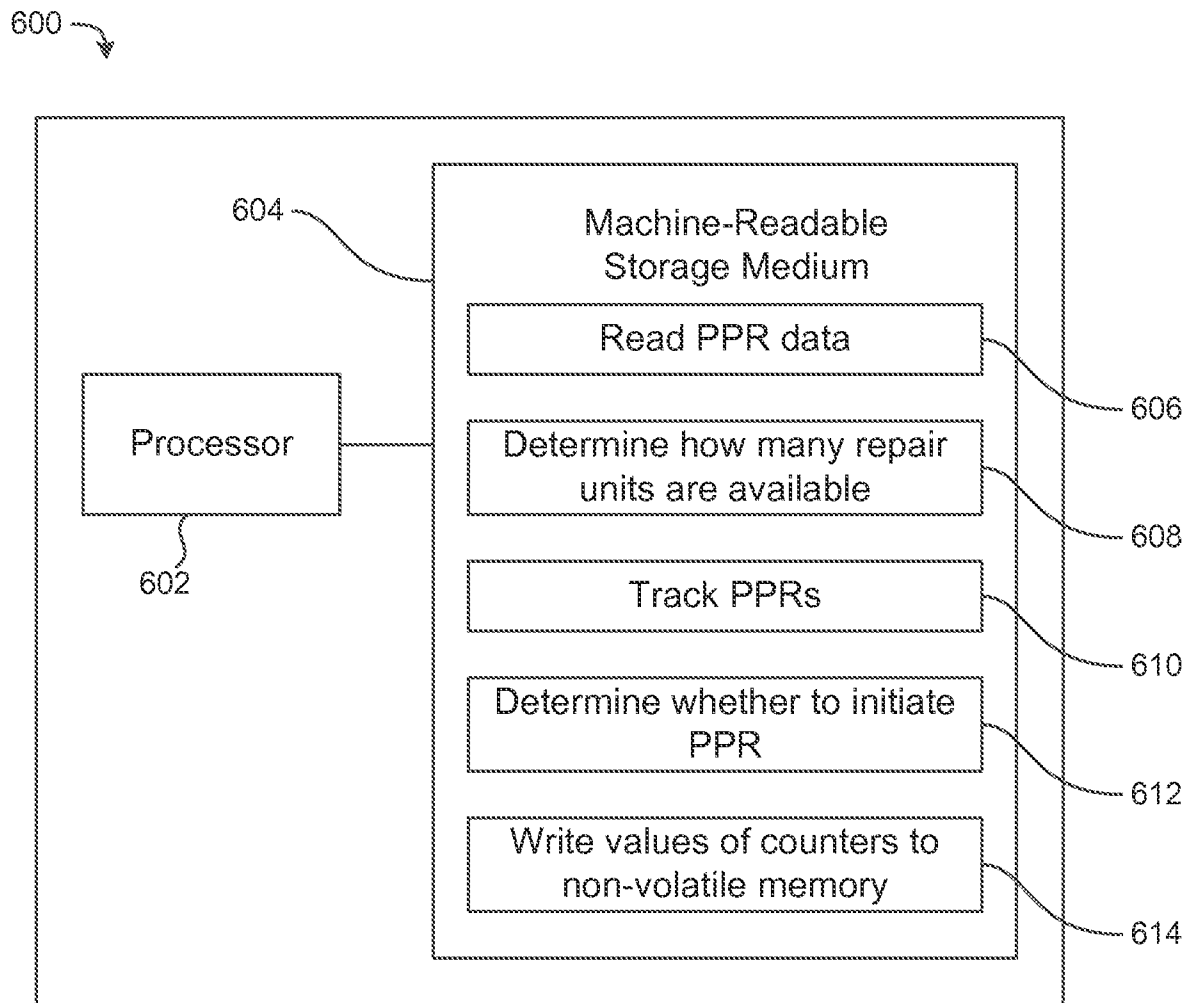
4/7

**FIG. 4**

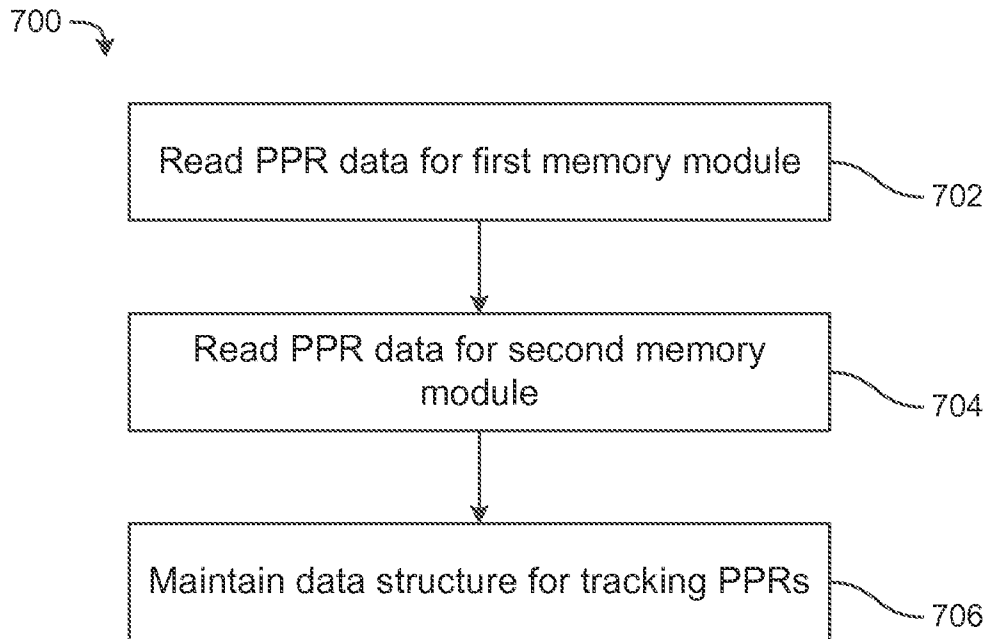
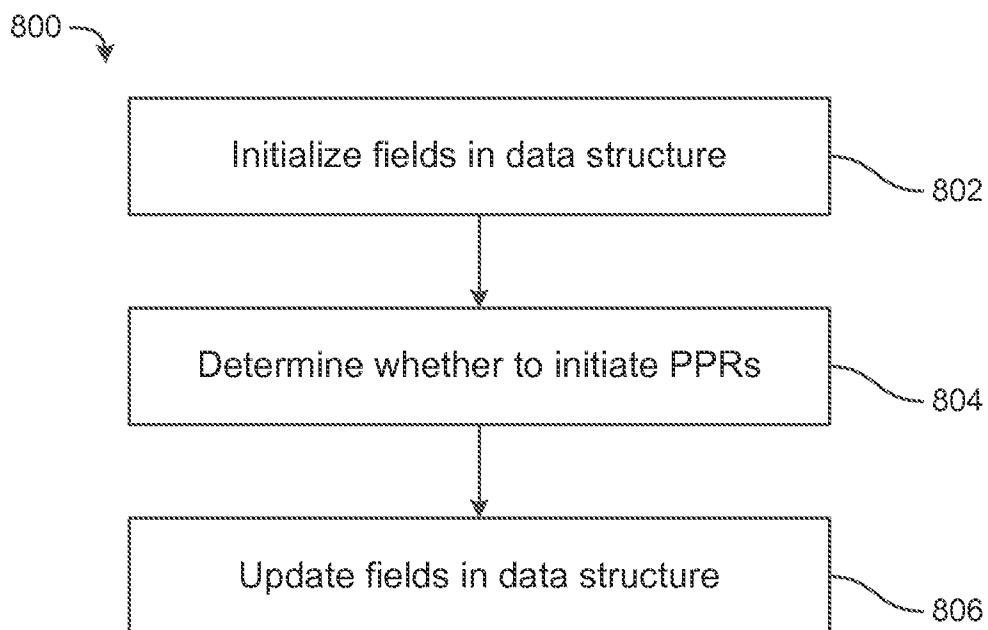
5/7

**FIG. 5**

6/7

**FIG. 6**

7/7

**FIG. 7****FIG. 8**

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2014/044354**A. CLASSIFICATION OF SUBJECT MATTER****G11C 29/00(2006.01)i, G11C 16/02(2006.01)i**

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G11C 29/00; G11C 7/10; G11C 7/00; H03M 13/05; G11C 8/06; G06F 11/00; G11C 16/02

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, volatile, repair, non-volatile, post, package, repair, PPR, boot, SPD, EEPROM

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2007-0058470 A1 (KLAUS NIERLE et al.) 15 March 2007	1-2,4,7-8,12-13
A	See paragraphs [0007], [0029], [0032], [0034]; and figures 1, 3.	3,5-6,9-11,14-15
Y	US 2011-0228614 A1 (IAN SHAEFFER et al.) 22 September 2011	1-2,4,7-8,12-13
A	See paragraph [0097]; and figure 10.	
A	US 2009-0217093 A1 (RAMON S. CO) 27 August 2009	1-15
A	See paragraph [0080]; and figure 10.	
A	US 2011-0296258 A1 (STUART SCHECHTER et al.) 01 December 2011	1-15
A	See paragraph [0029]; and figure 1.	
A	US 2007-0133323 A1 (HYUNG-JIK KIM et al.) 14 June 2007	1-15
A	See paragraph [0028]; and figure 1.	



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

14 April 2015 (14.04.2015)

Date of mailing of the international search report

15 April 2015 (15.04.2015)

Name and mailing address of the ISA/KR

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

Facsimile No. ++82 42 472 7140

Authorized officer

BYUN, Sung Cheal

Telephone No. +82-42-481-8262



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2014/044354

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2007-0058470 A1	15/03/2007	CN 1983453 A DE 102006043401 A1 US 7263019 B2	20/06/2007 24/05/2007 28/08/2007
US 2011-0228614 A1	22/09/2011	CN 101715593 A CN 101715593 B JP 2010-524089 A US 2008-080261 A1 US 2009-198924 A1 US 2010-146199 A1 US 2014-223068 A1 US 7562271 B2 US 7685364 B2 US 8108607 B2 US 8539152 B2 WO 2008-124503 A1	26/05/2010 12/06/2013 15/07/2010 03/04/2008 06/08/2009 10/06/2010 07/08/2014 14/07/2009 23/03/2010 31/01/2012 17/09/2013 16/10/2008
US 2009-0217093 A1	27/08/2009	US 2009-0217102 A1 US 7797578 B2 US 7797583 B2	27/08/2009 14/09/2010 14/09/2010
US 2011-0296258 A1	01/12/2011	US 8839053 B2	16/09/2014
US 2007-0133323 A1	14/06/2007	KR 10-0722771 B1 US 7486577 B2	22/05/2007 03/02/2009