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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
22 May 2014 (22.05.2014)

WIPO | PCT

(10) International Publication Number
WO 2014/078788 A1

(51) International Patent Classification:
G11C 16/06 (2006.01) *G11C 16/10* (2006.01)
G11C 16/34 (2006.01)

(21) International Application Number:
PCT/US2013/070561

(22) International Filing Date:
18 November 2013 (18.11.2013)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
13/680,507 19 November 2012 (19.11.2012) US

(71) Applicant: **SPANSION LLC** [US/US]; 915 Deguigne Dr., Sunnyvale, CA 94088-3453 (US).

(72) Inventors: **KIM, Yong, K.**; 3665 Vireo Avenue, Santa Clara, CA 95051 (US). **WONG, Keith, H.**; 4123 Arezzo Pointe Lane, San Jose, CA 95148 (US). **MCCLAIN, Mark, A.**; 12598 Dormouse Road, San Diego, CA 92129 (US).

(74) Agents: **FEATHERSTONE, Donald, J.** et al.; Sterne, Kessler, Goldstein & Fox P.L.L.C., 1100 New York Ave, Nw, Dc, Washington, DC 20005 (US).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) Designated States *(unless otherwise indicated, for every kind of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG).

Published:

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

(54) Title: DATA REFRESH IN NON-VOLATILE MEMORY

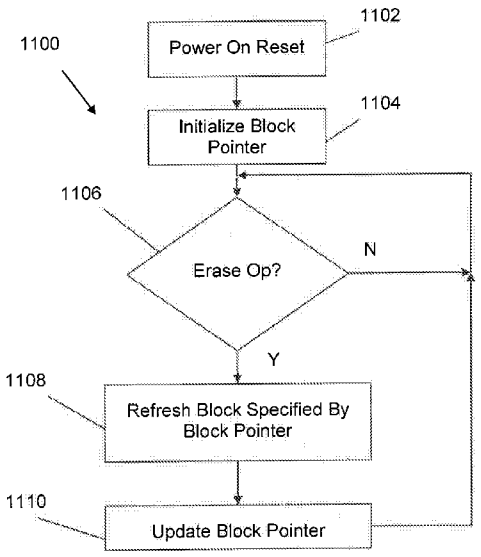


Fig. 11

(57) Abstract: A. method of reducing read errors in a non-volatile memory device that result from bit- line or word-line disturb conditions generated by erase operations includes selecting a subset of a memory array for refresh after each erase operation. A pointer to the refresh target section is updated as part of the method to direct the refresh operation to the appropriate subset of the memory array. Refresh may be performed subsequent to an erase operation or concurrently therewith. By distributing the time consumed by refresh operations over many erase operations so the relative refresh time for any one erase becomes small.

WO 2014/078788 A1

DATA REFRESH IN NON-VOLATILE MEMORY

Field of the Invention

[0001] The present invention relates generally to non-volatile memory, and more particularly relates to refreshing the content of non-volatile memory cells.

Background

[0002] Advances in semiconductor manufacturing processes, digital system architecture, and wireless infrastructure, among other things, have resulted in a vast array of electronic products, particularly consumer products, that drive demand for ever-increasing performance and density in non-volatile memory.

[0003] One means of increasing the performance and density of non-volatile memories such as flash memory, is to shrink the dimensions of floating gate transistors that are used in flash memories. It is well-recognized that shrinking the physical dimensions of the floating gate transistor also reduces the size of the floating gate itself and thus reduces the amount of charge that can be stored. When less charge storage is available, the memory cell becomes more sensitive to memory operations that unintentionally inject or remove charge, and thereby degrade the ability to store data successfully over many operational cycles. Mechanisms, other than erasing and programming, that affect the amount of charge stored on the floating gate of a flash memory cell include word line disturb error and bit line disturb error.

[0004] Word line and bit line disturb errors are similar to each other in that the voltages that appear across the terminals of a flash memory cell, i.e., across the terminals of a floating gate transistor, are such that electrons are caused to tunnel out of the floating gate, and typically into the drain. This unintended electron tunnelling causes a shift in the threshold voltage of the floating gate transistor, and consequently may change the data previously stored. This data degradation phenomenon occurs in a memory cell that has not been selected, but which shares a word-line and/or bit-line with one or more memory cells that have been selected for an erase operation.

[0005] In order to overcome the loss of data integrity caused by these unintended increases or decreases in stored charge, refresh operations have been used in flash

memories. Refresh operations read out data from a section of the flash memory array and re-program the corresponding memory cells with the same data.

[0006] Various schemes for refreshing the content of flash memory cells that are affected by word-line or bit-line disturb errors have been implemented in the past. Unfortunately, conventional refresh schemes that refresh all potentially affected memory cells at the same time, consume undesirable amounts of time and resources.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] Embodiments of the invention are described with reference to the accompanying drawings. In the drawings, like reference numbers indicate identical or functionally similar elements. Additionally, the left most digit(s) of a reference number identifies the drawing in which the reference number first appears.

[0008] Fig. 1 is a cross-sectional representation of an n-channel floating gate field effect transistor used in a non-volatile memory array such as flash memory.

[0009] Fig. 2 shows the structure of Fig. 1 during an erase operation in which electrons tunnel from the floating gate to the drain.

[0010] Fig. 3 shows the structure of Fig. 1 during a program operation in which electrons are injected onto the floating gate.

[0011] Fig. 4 shows the structure of Fig. 1 during a read operation in which some electrons are unintentionally injected onto the floating gate, in a process referred to as read disturb.

[0012] Fig. 5 shows the structure of Fig. 1 during an erase operation in an adjacent memory area that has a word line or a bit line in common with the illustrated transistor in which electrons tunnel from the floating gate as an unintended consequence of the erase operation in the adjacent memory area, in a process referred to as word line, or bit line disturb.

[0013] Fig. 6 is a schematic representation of a portion of a flash memory showing a target erase section, a section that shares word-lines with the target erase section (a word-line neighbor), a section that shares bit-lines with the target erase section (a bit-line neighbor), and an unrelated section of memory cells.

[0014] Fig. 7 shows a logical representation of an exemplary flash memory array that includes a plurality of big blocks, each big block including four blocks, and each block including sixteen sectors.

[0015] Fig. 8 shows a logical representation of a one big block that includes four blocks, each block having sixteen sectors, and further showing the state of a block counter associated with a refresh operation of each of the blocks in the big block.

[0016] Fig. 9 is a high-level block diagram of an exemplary flash memory device.

[0017] Fig. 10 is a flow diagram of an exemplary process in accordance with the present invention.

[0018] Fig. 11 is a flow diagram of another exemplary process in accordance with the present invention.

[0019] It is noted that the cross-sectional representations of various semiconductor structures shown in the figures are not necessarily drawn to scale, but rather, as is the practice in this field, drawn to promote a clear understanding of the structures, process steps, and operations which they are illustrating.

DETAILED DESCRIPTION

[0020] The following Detailed Description refers to accompanying drawings to illustrate exemplary embodiments consistent with the invention. References in the Detailed Description to "one exemplary embodiment," "an illustrative embodiment", "an exemplary embodiment," and so on, indicate that the exemplary embodiment described may include a particular feature, structure, or characteristic, but every exemplary embodiment may not necessarily include the particular feature, structure, or characteristic. Moreover, such phrases are not necessarily referring to the same exemplary embodiment. Further, when a particular feature, structure, or characteristic is described in connection with an exemplary embodiment, it is within the knowledge of those skilled in the relevant art(s) to affect such feature, structure, or characteristic in connection with other exemplary embodiments whether or not explicitly described.

[0021] The exemplary embodiments described herein are provided for illustrative purposes, and are not limiting. Other exemplary embodiments are possible, and modifications may be made to the exemplary embodiments within the spirit and scope of the invention. Therefore, the Detailed Description is not meant to limit the invention.

Rather, the scope of the invention is defined only in accordance with the following claims and their equivalents.

[0022] The following Detailed Description of the exemplary embodiments will so fully reveal the general nature of the invention that others can, by applying knowledge of those skilled in relevant art(s), readily modify and/or adapt for various applications such exemplary embodiments, without undue experimentation, without departing from the spirit and scope of the invention. Therefore, such adaptations and modifications are intended to be within the meaning and plurality of equivalents of the exemplary embodiments based upon the teaching and guidance presented herein. It is to be understood that the phraseology or terminology herein is for the purpose of description and not of limitation, such that the terminology or phraseology of the present specification is to be interpreted by those skilled in relevant art(s) in light of the teachings herein.

Terminology

[0023] The terms, chip, die, integrated circuit, semiconductor device, and microelectronic device, are often used interchangeably in the field of electronics. The present invention is applicable to all the above as these terms are generally understood in the field.

[0024] With respect to chips, it is common that power, ground, and various signals may be coupled between them and other circuit elements via physical, electrically conductive connections. Such a point of connection may be referred to as an input, output, input/output (I/O), terminal, line, pin, pad, port, interface, or similar variants or combinations. Although connections between and amongst chips are commonly made by way of electrical conductors, those skilled in the art will appreciate that chips and other circuit elements may alternatively be coupled by way of optical, mechanical, magnetic, electrostatic, and electromagnetic interfaces.

[0025] FET, as used herein, refers to a metal-oxide-semiconductor field effect transistor (MOSFET). An n-channel FET is referred to herein as an NFET. A p-channel FET is referred to herein as a PFET.

[0026] Floating gate transistor refers to a FET in which a stack is disposed over the channel region, where that stack has a first dielectric layer disposed over the channel region of the FET, a first charge storage material disposed over the first dielectric layer, a second dielectric layer disposed over the electrically conductive material, and a second

electrically conductive material disposed over the second dielectric layer. The first dielectric layer is referred to herein as the floating gate dielectric layer. Historically, and less accurately, the first dielectric layer was referred to as the tunnel oxide. The first charge storage material is referred to as the floating or charge trapping gate. The second dielectric layer is referred to herein as the control gate dielectric layer. Historically, the second dielectric layer was referred to as the interpoly oxide, but this term is misleading for floating gate transistors having a high-k dielectric material as the second dielectric layer, and a control gate made of metal, metal alloy, or a stack of metals or metal alloys (HKMG). The second electrically conductive material is referred to as the control gate. In this arrangement the floating gate is electrically insulated from any other electrical node.

[0027] Source/drain (S/D) terminals refer to the terminals of a FET, between which conduction occurs under the influence of an electric field, subsequent to the inversion of the semiconductor surface (i.e., the semiconductor surface disposed adjacent to the gate dielectric layer of the FET) under the influence of an electric field resulting from a voltage applied to the gate terminal of the FET. Generally, the source and drain terminals of a FET are fabricated such that they are geometrically symmetrical. With geometrically symmetrical source and drain terminals it is common to simply refer to these terminals as source/drain terminals, and this nomenclature is used herein. Designers often designate a particular source/drain terminal to be a "source" or a "drain" on the basis of the voltage to be applied to that terminal when the FET is operated in a circuit.

[0028] High-k dielectric refers to a material that has a dielectric constant greater than that of silicon dioxide.

[0029] Substrate, as used herein, refers to the physical object that is the basic workpiece that is transformed by various process operations into the desired microelectronic configuration. A substrate may also be referred to as a wafer. Wafers, may be made of semiconducting, non-semiconducting, or combinations of semiconducting and non-semiconducting materials.

[0030] The expression “refresh unit” refers to the number of memory cells that are refreshed by a refresh operation. The refresh unit may be described in terms of logical segments of a memory such as but not limited to a sector or a block.

Overview

[0031] A flash memory cell is a non-volatile memory cell that is typically formed from a floating gate transistor. Flash memory products generally include an array of flash memory cells. It is noted that the term array is not used to limit the invention in any way, but rather to convey that a plurality of flash memory cells are typically arranged into one or more logical groupings within a flash memory product.

[0032] Alternative non-volatile memory cells, such as those that trap charge within the gate dielectric layer, rather than on a floating gate, may also be used to produce flash memory products. The present invention is applicable to both memory products without regard to whether charge trapping transistors and floating gate transistors are used to implement the flash memory products.

[0033] In the flash memory it is common for a portion of the floating gate transistors in the array to be interconnected such that their control gates form a common node. That common node is referred to as a word-line. The word line is driven by a word-line driver circuit, which may apply a voltage to the control gates, that voltage having a magnitude dependent on whether these memory cells have been addressed, and on whether an erase, program, or read operation is to be performed. Flash memory arrays typically include many word-lines.

[0034] Similarly, in the flash memory array it is common for a portion of the floating gate transistors in the array to be interconnected such that their drains form a common node. This common node is referred to as a bit-line. A bit-line driver circuit may apply a voltage to the bit-line, that voltage having a magnitude dependent on whether a memory cell connected to the bit-line has been addressed, and on whether an erase, program, or read operation is to be performed. Flash memory arrays typically include many bit-lines.

[0035] In addressing, i.e., selecting, portions of a flash memory array, it is known that accessing the desired memory cell(s) can unintentionally place electrons on, or remove electrons from, the floating gate of non-selected memory cells, if those non-selected memory cells share a word-line and/or a bit-line with the selected memory cells. Adding or removing electrons will change the threshold voltage of the transistor of the memory cell, thus possibly moving the threshold voltage away from its targeted value.

[0036] In one particular example, during an erase operation, a selected set of memory cells are erased, and a neighboring set of memory cells that share a bit-line suffer from a

loss of electrons that were previously stored on the floating gates of those neighboring (i.e., bit-line sharing) memory cells. It is noted that a disturb can also happen during programming. In another example an erase algorithm, or process, may include a pre-programming operation such that, before the actual erase operation occurs, a programming operation is performed to place all selected memory cells in the programmed state before starting the erase operation. Both the programming operation and the erase operation can cause disturb in a neighboring set of memory cells.

[0037] A conventional approach to overcoming this undesired phenomenon is to perform a refresh operation. Unfortunately, conventional refresh operations can slow down the flash erase process. As explained in greater detail below, various embodiments of the present invention provide circuits and methods for reducing the amount of overhead associated with refreshing flash memory cells. Various embodiments advantageously maintain the integrity of data stored in the memory cells by directing the performance of refresh operations in an efficient manner.

[0038] Fig. 1 shows a cross-sectional representation of an illustrative n-channel floating gate field effect transistor **100** used in a non-volatile memory array such as a flash memory. A substrate **102**, typically, but not limited to, a silicon wafer, has formed therein a pair of source/drain (S/D) terminals **104**, **106**. That portion of substrate **102** laterally disposed between source **104** and drain **106** is referred to herein as a channel region **103**. As can be seen in Fig. 1, source **104** and drain **106** are aligned to a stack that includes a gate dielectric layer **108**, a floating gate **110**, a dielectric layer **112**, and a control gate **114**. Floating gate **110** is electrically insulated from channel region **103** by gate dielectric layer **108**, and is electrically insulated from control gate **114** by dielectric layer **112**. In describing the operation of floating gate transistor **100**, S/D terminal **104** is designated to be source terminal **104**, and S/D terminal **106** is designated to be drain terminal **106**. The threshold voltage of floating gate transistor **100** can be modified by increasing or decreasing the amount of charge stored on floating gate **110**. Electrons are typically placed on the floating gate by hot electron injection, and removed by tunneling. By convention, removing electrons is referred to as erasing, and adding electrons is referred to as programming. Those skilled in the art will recognize that Fig. 1 is illustrative only, and that other device structures that implement the functionality of the illustrated floating gate transistor are possible. By way of example, and not limitation,

the control gate may wrap around the vertical sides of the floating gate, with both gates separated by a dielectric material. By way of further example, and not limitation, dielectric layer **112** and control gate **114** may be comprised respectively of a high-k dielectric material and a metal, metal alloy, or stack of metals and/or metal alloys.

[0039] Fig. 2 shows the structure of Fig. 1 during an illustrative erase operation in which electrons tunnel from the floating gate to the drain. In this example, to erase floating gate transistor **100**, source terminal **104** is floated, i.e., isolated from all voltage sources, a voltage source V_{CG} is applied to control gate **114**, and a voltage source V_D is applied to drain **106** such that a voltage difference of a predetermined magnitude is obtained between V_{CG} and V_D. The applied voltages are such that the control gate is not attracting electrons and the drain is attracting electrons. This arrangement causes electrons stored on floating gate **110** to tunnel through floating gate dielectric **108**, thereby reducing the amount of negative charge on floating gate **110**, which in turn makes the threshold voltage of floating gate transistor **100** lower, i.e., able to turn on at a lower control-gate-to-source voltage.

[0040] It is noted that in alternative arrangements, the voltage in the channel region of the floating gate transistor, and/or the voltage in the well of the floating gate transistor, can also affect the erase operation. The present invention is not limited to any particular set of voltages or the application of those voltages on specific terminals of the floating gate transistors. Rather, the application of voltage differences between various terminals in any suitable combination to perform the desired erase, program, and refresh operations are contemplated.

[0041] Fig. 3 shows the structure of Fig. 1 during a program operation in which electrons are injected onto the floating gate. To program floating gate transistor **100**, a voltage source Vs is applied to source terminal **104**, voltage source V_{CG} is applied to control gate **114**, and voltage source V_D is applied to drain **106** such that both the control-gate-to-source voltage and the drain-to-source voltages are predetermined positive values. This arrangement causes electrons to be accelerated in channel region **103** between source **104** and drain **106** under the influence of the drain-to-source electric field. The accelerated electrons become sufficiently energetic that some of them have enough energy, after a collision with one or more atoms in the crystal lattice of channel region **103**, to pass through floating gate dielectric layer **108** and become trapped in floating gate **110**. The

increased amount of negative charge on floating gate **110** makes the threshold voltage of floating gate transistor **100** higher, i.e., requiring a higher control-gate-to-source voltage to turn on.

[0042] Fig. 4 shows the structure of Fig. 1 during a read operation in which some electrons are unintentionally injected onto the floating gate, in a process referred to as read disturb. In this example, to perform a read operation, floating gate transistor **100** is configured similarly to the programming operation shown in Fig. 3, except that the magnitude of the voltages applied to control gate **114** and drain **106** are both less than those applied during programming. Even though the voltages are significantly lower during a read operation than during a program operation, it is still possible for some electrons to obtain sufficient energy to be injected through floating gate dielectric **108** and into floating gate **110**. This unintended charge injection increases the threshold voltage of floating gate transistor **100**, and over some number of read operations this process can inject enough charge to change the state of the memory cell. This effect is sometimes referred to as a read disturb error.

[0043] Fig. 5 shows the structure of Fig. 1 during an erase operation in an adjacent memory area that has a word line or a bit line in common with the illustrated transistor in which electrons tunnel from the floating gate as an unintentional consequence of the erase operation in the adjacent memory area, in a process referred to as word line, or bit line disturb. In this context, "adjacent memory area" refers to a section of the memory that shares at least one word line or at least one bit line with the section of memory that is the target of the erase operation.

[0044] Fig. 6 is a schematic representation of a portion of a flash memory **600** showing a target erase section **602**, a section **606** that shares word-lines with target erase section **602** (referred to as a word-line neighbor), a section **604** that shares bit-lines with target erase section **602** (referred to as a bit-line neighbor), and a section **608** that does not share word-lines or bit-lines with target erase section **602** (referred to as an unrelated neighbor).

[0045] Still referring to Fig. 6, in an illustrative erase operation, the word-lines and bit-lines of erase target section **602** are set to the appropriate voltages so that electrons leave the floating gates by tunneling through the respective floating gate dielectric layers. Only the memory cells in erase target section **602** are intended to be erased. In an exemplary arrangement, bit-line neighbor **604**, which shares bit-lines with target erase section **602**, is

exposed to the relatively high bit line voltage used during the erasing of section **602**. In erasing the voltage differential between the control gate of a memory cell and a bit line of the memory cell is increased to produce electron tunneling from the floating gate to the bit-line (i.e., drain). Although the voltage differential between the control gate of memory cells in bit-line neighbor section **604** and the shared bit-lines is not as great as that experienced in erase target section **602**, it is great enough to create a disturb condition that over repeated occurrences leads to read disturb errors. To overcome this problem it is necessary to refresh the potentially charge degraded memory cells. The disturb resulting from any one erase operation is small as compared to the level of disturb needed to cause a read error, generally requiring thousands of erase operations to cause a read error. It is therefore not necessary to refresh potentially disturbed memory cells after each erase operation. It is only necessary to refresh each potentially affected memory cell once before the number of erase cycles begins to approach the disturb level that could create a read error. This allows the refresh operation to be done on one small group of potentially affected memory cells following each erase operation and moving to a new group of memory cells for each erase operation such that the refresh work on the entire memory cell array is distributed across many erase cycles. This reduces the time and energy spent on refresh related to each erase operation so as to not significantly increase the duration or power consumption of each individual erase operation.

Illustrative Embodiment

[0046] Several illustrative embodiments are presented below and described in the context of a 16Mb flash memory array that is divided into eight Big Blocks, with each Big Block having four blocks, and each block having sixteen sectors.

[0047] Fig. 7 is a high-level block diagram showing an illustrative flash memory array **700** logically organized as a collection of eight "big blocks" **702**. Each big block **702** is shown to be organized as a collection of four blocks **704a**, **704b**, **704c**, **704d**. Each of blocks **704a**, **704b**, **704c**, **704d** is logically organized as 16 sectors. Each of blocks **704a**, **704b**, **704c**, **704d** may be referred to, respectively, as Block 0, Block 1, Block 2 and Block 3. In this illustrative embodiment, a common local bitline is shared in the big block, and the big block is thus susceptible to disturb errors. The present invention is not limited to any particular logical arrangement or physical grouping of memory cells.

[0048] Fig. 8 shows a single big block **702**, and its four constituent blocks, Block 0, Block 1, Block 2 and Block 3. Further shown, adjacent to each of Block 0, Block 1, Block 2 and Block 3 is a unique state of a block counter (not shown) The arrows in Fig. 8 indicate that, in this illustrative embodiment, the state of the block counter transitions through the following states: 00, 01, 10, 11, and then back to state 00. Various embodiments use the state of the block counter to specify which block is to be refreshed. The present invention is not limited to any particular implementation of the block counter. In various illustrative embodiments the block counter may change state by incrementing, decrementing, shifting its contents, acting as a polynomial counter, being loaded with a value from other circuitry including but not limited to a micro-controller, or any other suitable means of changing state.

[0049] Fig. 9 is a high-level block diagram of an illustrative flash memory device **902**. Flash memory device **902** includes a memory array, which in turn includes a plurality of big blocks **702**; control circuits **906**; and interface circuits **908**. Memory array **902** is coupled to both control circuits **906** and interface circuits **908**; and control circuits **906** are further coupled to interface circuits **908**. Memory array **902** typically includes well-known row and column address decoding circuitry (not shown). Control circuits **906** includes the circuitry necessary to perform the various operations of memory device **902** such as the well-known erase, program, and read operations. Typical embodiments include a microcontroller that operates responsive to stored instructions. Such microcontrollers within flash memory devices are used to control various operations that often include many steps. Control circuits **906**, further include a block counter **910**. Interface circuits **908** include the circuitry necessary to perform the various well-known input/output buffering of such flash memory devices.

Process

[0050] Various embodiments of the present invention distribute refresh time over many erase operations so that the refresh time for any one erase is small compared to the brute force method of refreshing an entire big block responsive to every erase operation. In some embodiments, the erase process includes both a pre-programming operation and an erase operation.

[0051] Various embodiments use sequential logic or a microcontroller embedded with a flash memory device to execute the refresh algorithm every time the erase procedure is

done. In some embodiments the refresh procedure is part of the same sequential logic operation or microcontroller program routine as the erase procedure.

[0052] Increased benefits from the present invention occur as the big block is divided into smaller refresh units. Refresh unit refers to that portion of the memory that is refreshed during the refresh cycle initiated in response to an erase cycle. The smaller the refresh unit becomes, the shorter is the refresh time that is added to each erase/refresh cycle pair. For example, an individual word line could be refreshed.

[0053] It is noted that one design consideration in implementing various embodiments is the ratio of refresh units to the program/erase cycle limit of the memory device without refresh. Without performing refresh operations it is known that enough disturb will accumulate within the program/erase cycle limit of the memory device to cause read errors in sectors that have not been erased/refreshed. As long as the number of refresh units (e.g., word lines in a big block) are low enough in relation to the program/erase cycle limit, eventually all refresh units will be refreshed before reaching the program/erase cycle limit, thereby resetting the limit on the shared bit line disturb effect.

[0054] There are various ways of selecting which refresh unit to perform the refresh upon, following each erase operation. When the ratio of refresh units, in a big block, to the program/erase cycle limit of the memory device is low, e.g., 1/1000, the selection of the refresh unit can be random. This means there will be many more refresh operations than erase cycles allowed and the probability that all refresh units will be refreshed, via random selection, before the erase cycle limit is reached, is very near 100%. It is noted that there is no need to refresh within the sector just erased. So, if the selected refresh unit is in the just erased sector, the refresh operation can either be skipped or directed to another refresh unit. Also, it is only necessary to refresh within the big block where a sector is erased.

[0055] One method for selecting refresh units is to provide one counter with a count range equal to the number of refresh units in a big block. The present invention is not limited to any particular number of refresh units per big block. The counter can be incremented following each erase operation to any big block in the device. If all erase operations are directed to one big block, then the refresh units will be refreshed in sequence. If erase operations are distributed among big blocks, then the refresh unit selection will become more random with respect to any one big block. Thus effectively

moving randomly through refresh units in all big blocks because erases tend to occur in different big blocks.

[0056] However, the ratio of refresh units to program/erase cycle limit cycles can be lowered by increasing the certainty that all refresh units are covered within the program/erase cycle limit. This may be achieved by providing a counter for each big block, or by adding a bit in each refresh unit that can be programmed to mark it as refreshed. Either method ensures linear progress through the refresh units by tracking within each block rather than relying on a single counter per memory device that is incremented by any erase.

[0057] In the illustrative embodiment of Fig. 8, the state of a two-bit binary counter is shown. In this embodiment the counter increments by 1 after each refresh. Those skilled in the art and having the benefit of the present disclosure will understand that any suitable means or apparatus for producing the required number of unique states can be used. By way of example and not limitation, the block counter may be a binary counter of three or more bits which resets to a beginning state after counting through an adequate number of unique states. In some embodiments the binary counter counts up, counts down, or counts up and then down, or down and then up. In another embodiment the block counter is a polynomial counter rather than a binary counter. In another embodiment, the block counter is a storage register whose contents are loaded from the output of other circuitry, for example, an on-chip microcontroller.

[0058] It is noted that the block counter may alternatively be referred to as a counter, a refresh counter, a block pointer, a refresh pointer, a pointer, a block refresh address register, a refresh address register, and similar terms or expressions. No limitation is intended by the choice of terminology for this functionality. The functional requirement of this illustrative embodiment is that a block counter be able transition through an adequate number of states such that all of the subsections of the memory are eventually refreshed. Any suitable implementation of the refresh counter functionality may be used, including but not limited to a binary counter, a polynomial counter, a shift register, a loadable non-counting register, a micro-controller.

[0059] The state of the block counter indicates which one block "n" to refresh after an erase operation. In this embodiment the block to be refreshed is within the same big block as the erased sector, the counter increments by 1, and wraps from state 11 to state

00 (as shown in Fig. 8). In this way four erase operations (either sector erase or block erase) in a big block will result in the entire big block being refreshed. This is because, in this illustrative embodiment, a block refresh in a big block is performed every time a sector or block in the big block is erased.

[0060] The present invention is not limited to the 16 sectors/block and 4 blocks/big block of the illustrative embodiments described above. Some memory devices may have more or fewer sectors per block, and/or more or fewer blocks per big block. However, embodiments of the present invention provide an adequate number of refreshes such that data integrity is maintained in the memory cells by refreshing the entire array, block by block, before the critical number of program/erase cycles for that memory array has occurred.

[0061] In some embodiments, every erase operation, anywhere in the memory device, increments a single block counter in the device, which helps to randomize the location of refresh in any particular big block. Such a configuration helps to avoid the situation where the power cycles (i.e., power on/power off) occur more frequently than erase commands thus always resetting the block counter to zero, or some known state, before all four blocks are refreshed. In alternative embodiments, the block counter is not reset by power cycles but rather powers up in a random state. In further alternative embodiments, the block counter is loaded with a random seed value upon a power on reset sequence.

[0062] Fig. 10 shows a flow diagram of a process **1000** in accordance with the present invention. At a step **1002** an erase operation is performed that includes exposing a non-targeted portion of a memory block to a voltage differential large enough to potentially accelerate data decay (i.e., a disturb condition). In this illustrative embodiment, the erase operation is performed responsive to the receipt of an erase command, but the invention is not so limited. The erase operation may be directed to a sector or to a block, but the invention is not limited to erasing any particular number of memory cells. Moreover, references to an erase operation also comprehend pre-programming and post-erase operations, since any or all of these operations can lead to disturb errors. At a step **1004** a refresh operation is performed in which a block specified by a block pointer is performed. At a step **1006** the block pointer is incremented such that it specifies the next block to be refreshed responsive to an erase operation.

[0063] Fig. 11 shows a flow diagram of a process **1100** in accordance with the present invention. At a step **1102** a power-on reset is shown as the beginning of process **1100**. At a step **1104** a block pointer is initialized. In this illustrative embodiment, the block pointer is a 2-bit binary counter that is reset to zero. At a step **1106** if no erase operation has been performed then control remains at step **1106**. If an erase operation has been performed then at a step **1108** a refresh operation is performed on a block specified by the block pointer. At a step **1110** the block pointer is updated to a new state. The new state of the block pointer specifies the next block to be refreshed.

[0064] With respect to the embodiment illustrated in Fig. 11, it is noted that alternative embodiments may not require that the block pointer be reset after a power-on reset, or after any other reset sequence. Alternative embodiments may increment the block pointer prior to performing the block refresh operation rather than afterwards. It is further noted that the present invention is not limited to refreshing any particular number of memory cells during a refresh operation. Further, embodiments may refresh one or more bits concurrently during a refresh operation.

[0065] Flash memory devices are known in this field that are responsive to both a sector erase command and to a block erase command. Various embodiments of the present invention may respond to the sector erase command and the block erase command in the same way. Still other embodiments may respond differently in order to provide greater efficiency. For example, in the case of a block erase, the entire block is erased rather than one sector, thus there are no other sectors in the block that need to be refreshed. If the block pointer specifies refresh of the block that has just been erased then in some embodiments the refresh operation can be skipped, while in other embodiments the pointer is incremented first so that it points to a block other than the one that was just erased.

[0066] In some embodiments refresh activity in one block takes place concurrently with erase activity in another block. One constraint on such embodiments is the total current draw on the charge pumps used in a flash memory device. However, if a flash memory device design includes charge pumps robust enough to provide the required amount of current, and further includes separate microcontroller code threads to manage the concurrent operations it would be possible. It is noted that the erase operation would put

some disturb on the other blocks so, the timing of read verify in the refresh block would have to be managed taking the erase timing into account.

[0067] In some embodiments the refresh activity begins with the decoding of the erase command. In some embodiments the erase operation begins first and the refresh activity begins before the erase operation is complete.

[0068] It is noted that various embodiments may be used to provide refreshing of the data in the memory cells regardless of the electrical conditions that have caused a degradation in the memory cell's pre-disturbed charge state.

[0069] It is noted that various embodiments may perform an erase operation rather than a program operation to accomplish the desired data refresh operation.

CONCLUSION

[0070] It is to be appreciated that the Detailed Description section, and not the Abstract of the Disclosure, is intended to be used to interpret the claims. The Abstract of the Disclosure may set forth one or more, but not all, exemplary embodiments of the invention, and thus, is not intended to limit the invention and the subjoined Claims in any way.

[0071] It will be apparent to those skilled in the relevant art(s) that various changes in form and detail can be made therein without departing from the spirit and scope of the invention. Thus the invention should not be limited by any of the above-described exemplary embodiments, but should be defined only in accordance with the subjoined Claims and their equivalents.

What is Claimed is:

1. A method of distributing across multiple erase operations, refreshing of content of non-volatile memory cells in an integrated circuit, comprising:

detecting an erase operation;

refreshing at least one group of non-volatile memory cells, the at least one group specified by a state of a pointer; and

changing the state of the pointer;

wherein the at least one group of non-volatile memory cells is less than all of the non-volatile memory cells.

2. The method of Claim 1, wherein detecting the erase operation comprises:

detecting a receipt of an erase command at the integrated circuit.

3. The method of Claim 1, wherein detecting the erase operation comprises:

detecting an occurrence of an erase command at the integrated circuit.

4. The method of Claim 1, further comprising:

performing pre-programming operation prior to an erase operation.

5. The method of Claim 4, wherein refreshing occurs concurrently with the erase operation.

6. The method of Claim 4, wherein refreshing occurs subsequent to the erase operation.

7. The method of Claim 4, wherein the erase operation is a block erase, and the pointer specifies refreshing an erased block, and further comprising:

skipping the refresh operation.

8. The method of Claim 4, wherein the erase operation is a block erase, and the pointer specifies refreshing an erased block, further comprising:

incrementing the pointer and subsequently refreshing a block specified by the pointer.

9. A method of refreshing the content of an array of non-volatile memory cells, the array organized to have a plurality of blocks, and each block having a plurality of sectors, comprising:

(a) erasing a sector of non-volatile memory cells;

(b) refreshing one block of the plurality of blocks where the block that is refreshed is selected based, at least in part, on a state of a block counter; and

(c) changing the state of the block counter.

10. The method of Claim 9, further comprising:

repeating steps (a) through (c).

11. The method of Claim 10, further comprising:

initializing the state of the block counter.

12. The method of Claim 10, wherein erasing and refreshing are controlled by a program executed by a microcontroller embedded in an integrated circuit with the array of non-volatile memory cells.

13. The method of Claim 10, wherein an erase operation occurs prior refreshing, and a pre-programming operation occurs prior to the erase operation.

14. The method of Claim 10, wherein erasing occurs concurrently with refreshing.

15. The method of Claim 10, wherein changing the state of the block counter comprises:

incrementing the contents of the block counter.

16. The method of Claim 10, wherein changing the state of the block counter comprises:

shifting the contents of the block counter.

17. A computer-readable storage device having instructions stored thereon that, when executed by microcontroller embedded in non-volatile memory integrated circuit including an array of non-volatile memory cells, the array organized to have a plurality of blocks, and each block having a plurality of sectors, causes the microcontroller to perform operations comprising:

(a) erasing a portion of the non-volatile memory cells;

(b) refreshing one block of a plurality of blocks where the block that is refreshed is selected based, at least in part, on a state of a block counter; and

(c) changing the state of the block counter; and

(d) repeating steps (a) through (c).

18. The computer-readable storage device of Claim 17, wherein changing the state of the block counter comprises incrementing.

19. The computer-readable storage device of Claim 17, wherein changing the state of the block counter comprises shifting the contents of the block counter.

20. The computer-readable storage device of Claim 17, further comprising:

(e) initializing the state of the block counter

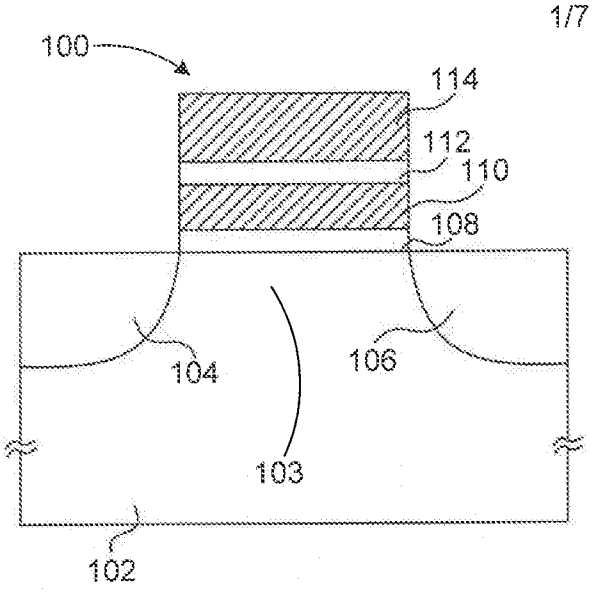


FIG. 1

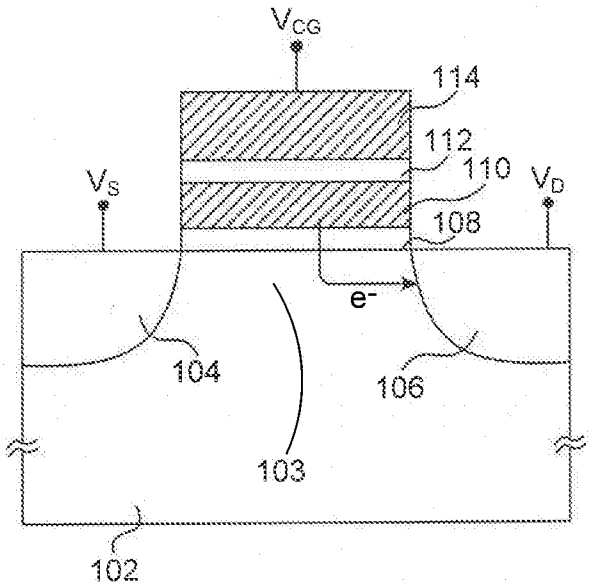


FIG. 2
Erase/Tunnelling

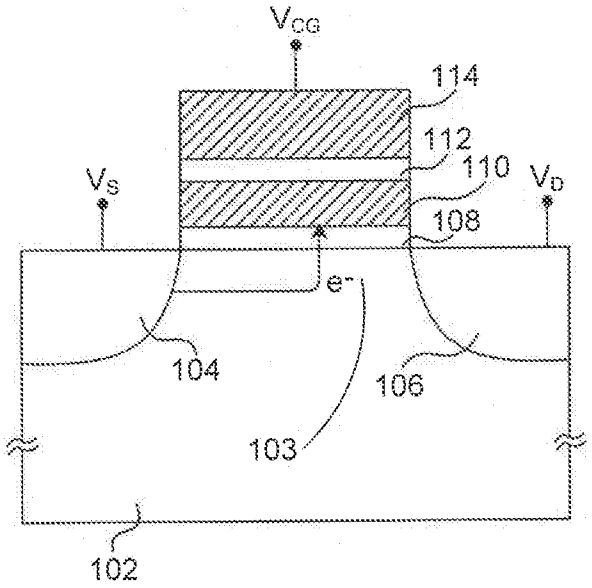


FIG. 3
Program/Injection

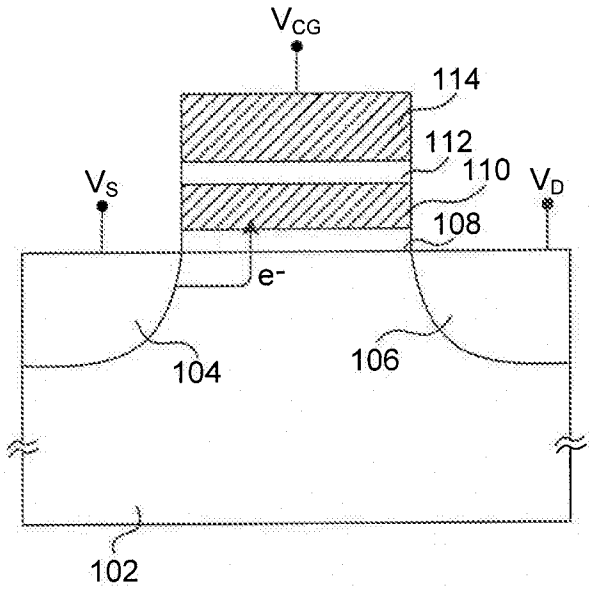


FIG. 4
Read/Unintended Injection

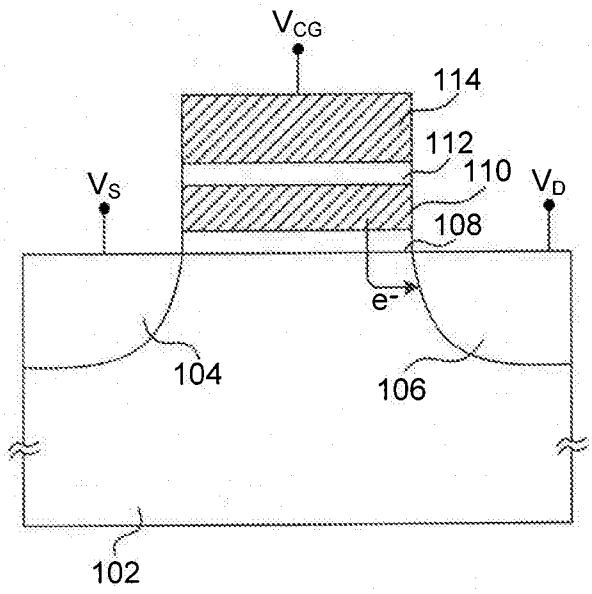
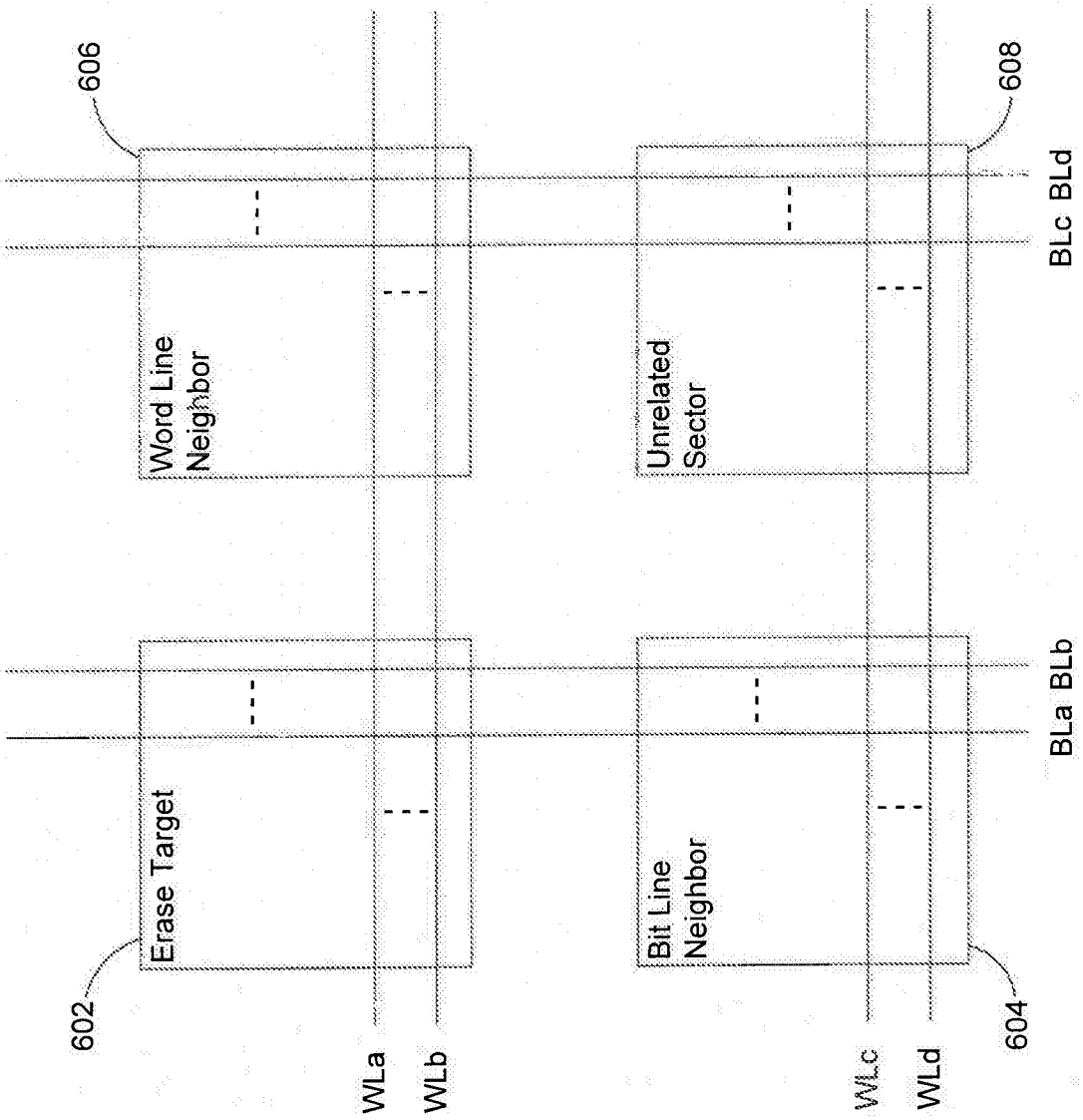


FIG. 5
Unaccessed Device Sharing Word Line or Bit Line with Erase Sector

FIG. 6


606
608
Word Line Neighbor
Unrelated Sector
BLc BLd
BLa BLb
Erase Target
Bit Line Neighbor
602
604
WLa
WLb
WLc
WLd

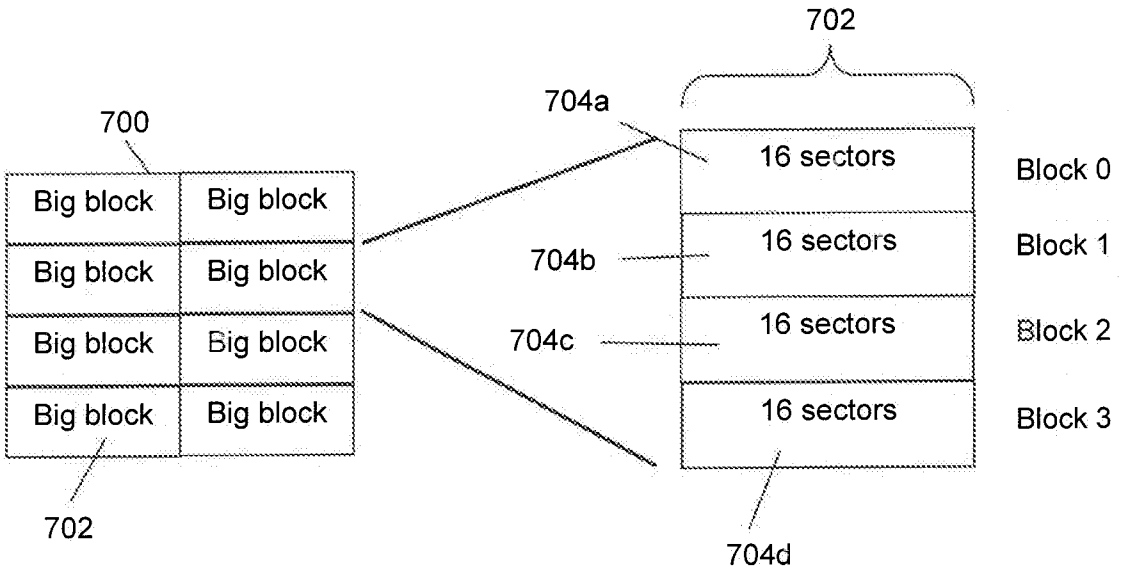

702
704a
700
16 sectors
Block 0
Big block
Big block
16 sectors
Block 1
704b
Big block
Big block
16 sectors
Block 2
704c
Big block
Big block
16 sectors
Block 3
Big block
Big block
702
704d

Fig. 7

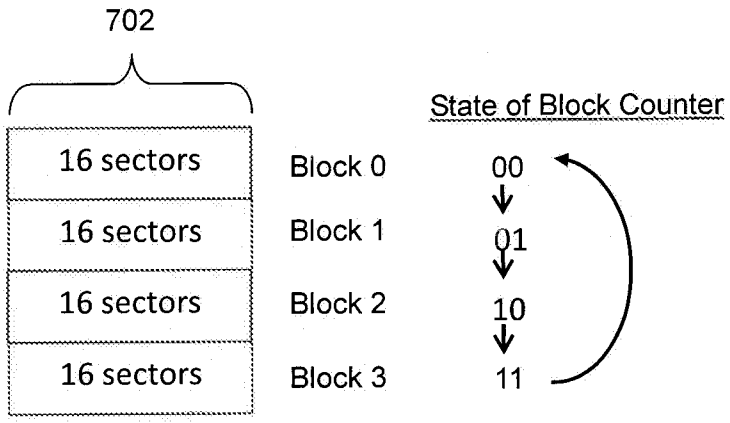

702
State of Block Counter
16 sectors
Block 0
00
16 sectors
Block 1
01
16 sectors
Block 2
10
16 sectors
Block 3
11

Fig. 8

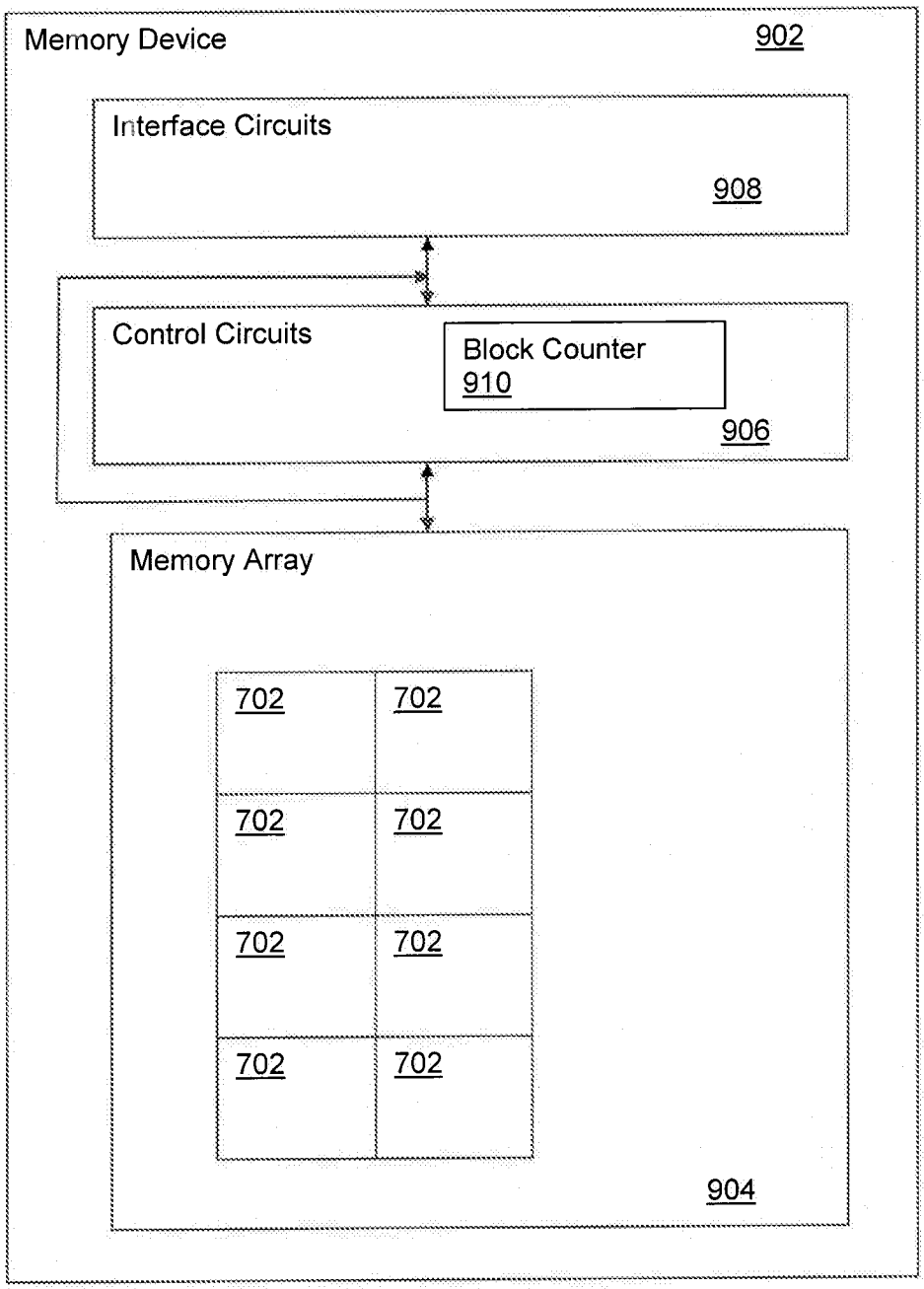

Memory Device
902
Interface Circuits
908
Control Circuits
Block Counter
910
906
Memory Array
702
702
702
702
702
702
702
702
904

Fig. 9

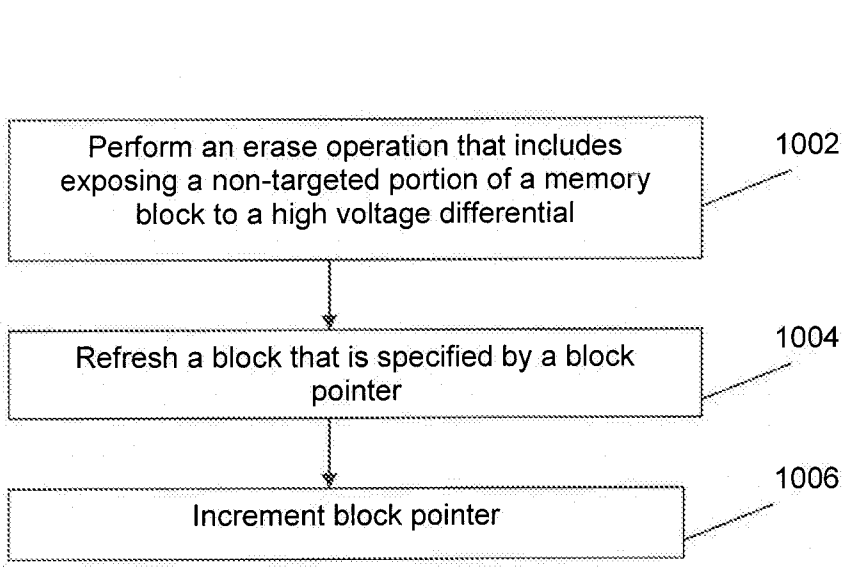

1000
Perform an erase operation that includes exposing a non-targeted portion of a memory block to a high voltage differential
1002
Refresh a block that is specified by a block pointer
1004
Increment block pointer
1006

Fig. 10

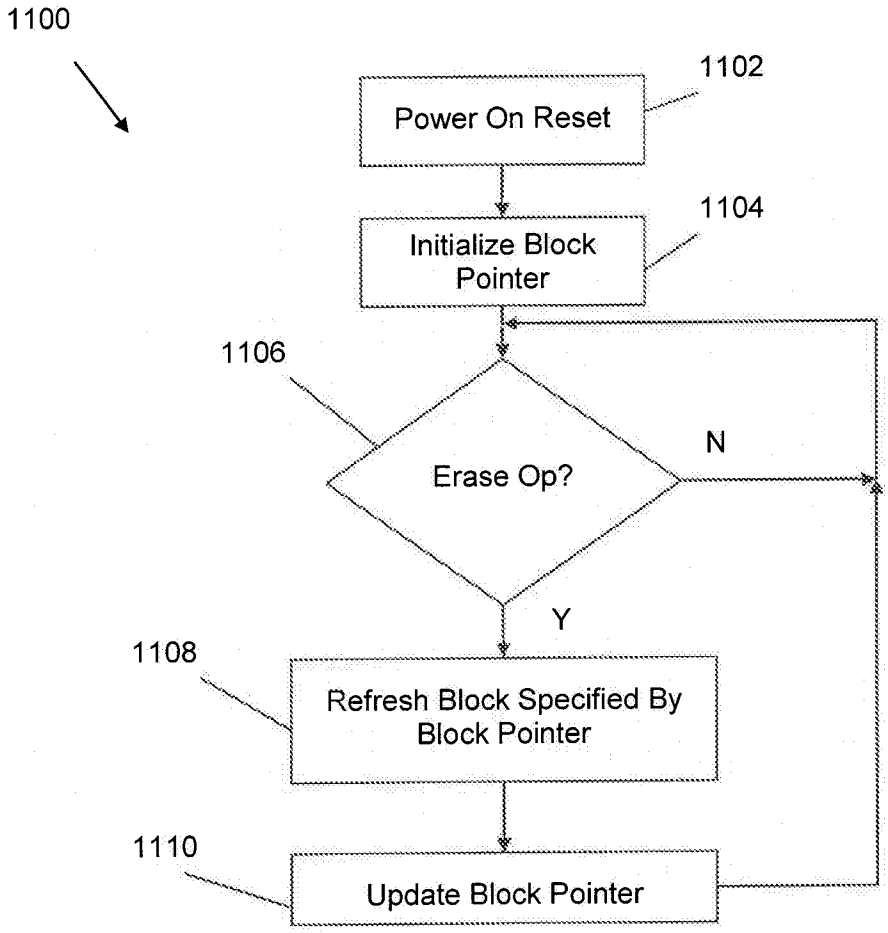

1100
1102
Power On Reset
1104
Initialize Block Pointer
1106
Erase Op?
N
Y
1108
Refresh Block Specified By Block Pointer
1110
Update Block Pointer

Fig. 11

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2013/070561

A. CLASSIFICATION OF SUBJECT MATTER

G11C 16/06(2006.01)i, G11C 16/34(2006.01)i, G11C 16/10(2006.01)i

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G11C 16/06; G06F 12/02; G11C 7/00; G06F 12/00; G11C 29/00; G11C 16/34; G11C 16/10

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

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords:memory, refresh, erase, pointer, counter, and similar terms.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 2011-0093671 A1 (SHUBA SWAMINATHAN) 21 April 2011 See paragraphs 19, 22, 25-26, 30; figures 2-3; and claim 97.	1-20
Y	US 8161355 B2 (STEPHEN FUNG et al.) 17 April 2012 See column 7, lines 32-36; column 8, lines 31-34; and figures 4-5.	1-20
A	US 7936610 B1 (ROBERT MELCHER et al.) 03 May 2011 See column 8, line 63 - column 11, line 13; and figures 5-7.	1-20
A	US 7010644 B2 (TERRY L. GILTON) 07 March 2006 See column 7, line 41 - column 10, line 35; and figures 4-5.	1-20
A	US 2007-0258306 A1 (JAMES E. PIEKELS et al.) 08 November 2007 See paragraphs 26-31, 45-47; and figures 1-2, 5.	1-20

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

☒ See patent family annex.

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

Date of the actual completion of the international search	Date of mailing of the international search report
20 February 2014 (20.02.2014)	**21 February 2014 (21.02.2014)**

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

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

Form PCT/ISA/210 (second sheet) (July 2009)

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.
PCT/US2013/070561

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2011-0093671 A1	21/04/2011	US 2003-0046487 A1	06/03/2003
		US 2005-0201153 A1	15/09/2005
		US 2008-0091873 A1	17/04/2008
		US 6931480 B2	16/08/2005
		US 7325089 B2	29/01/2008
		US 7870330 B2	11/01/2011
US 8161355 B2	17/04/2012	US 2010-0205504 A1	12/08/2010
		WO 2010-093441 A1	19/08/2010
		WO 2010-093441 A8	24/02/2011
US 7936610 B1	03/05/2011	None	
US 7010644 B2	07/03/2006	AT 517417 T	15/08/2011
		AT 557398 T	15/05/2012
		AU 2003-268251 A1	19/03/2004
		CN 100483546 C	29/04/2009
		CN 1689111 A	26/10/2005
		EP 1540657 A2	15/06/2005
		EP 1540657 B1	20/07/2011
		EP 2293305 A1	09/03/2011
		EP 2293305 B1	09/05/2012
		JP 04891544 B2	07/03/2012
		JP 2005-537598 A	08/12/2005
		KR 10-0626933 B1	20/09/2006
		KR 10-2005-0057034 A	16/06/2005
		TW 263221 B	01/10/2006
		TW I263221 B	01/10/2006
		US 2004-0044841 A1	04/03/2004
		US 2006-0104142 A1	18/05/2006
		US 2007-0258308 A1	08/11/2007
		US 2009-0257299 A1	15/10/2009
		US 2010-0262791 A1	14/10/2010
		US 7307908 B2	11/12/2007
		US 7564731 B2	21/07/2009
		US 7768861 B2	03/08/2010
		US 7944768 B2	17/05/2011
		WO 2004-021359 A2	11/03/2004
		WO 2004-021359 A3	24/02/2005
US 2007-0258306 A1	08/11/2007	EP 2016589 A2	21/01/2009
		JP 2009-536423 A	08/10/2009
		NO 20085065 A	29/01/2009
		US 7447096 B2	04/11/2008
		WO 2007-130187 A2	15/11/2007
		WO 2007-130187 A3	24/04/2008