



- (51) **International Patent Classification:**
G11C 16/10 (2006.01) *G11C 16/04* (2006.01)
- (21) **International Application Number:**
PCT/US2013/056826
- (22) **International Filing Date:**
27 August 2013 (27.08.2013)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
13/599,208 30 August 2012 (30.08.2012) US
- (71) **Applicant:** MICRON TECHNOLOGY, INC. [US/US];
Ms 525, 8000 South Federal Way, Boise, Idaho 83716 (US).
- (72) **Inventors:** GOLDMAN, Matthew; 1627 Zerlang Courth,
Folsom, California 95630 (US). HELM, Mark A.; 22
Eastridge Dr., Santa Cruz, California 95060 (US). PATEL,
Jaydip B.; 1200 Creekside Drive, Apt. 2634, Folsom,
California 95630 (US). RYAN, Thomas F.; 170 Rebecca
Way, Folsom, California 95630 (US).
- (74) **Agent:** LEFFERT, Thomas, W.; Leffert Jay & Polglaze,
P. A., P.O. Box 2230, Minneapolis, Minnesota 55402-
0230 (US).

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

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

Published:

- with international search report (Art. 21(3))
- before the expiration of the time limit for amending the claims and to be republished in the event of receipt of amendments (Rule 48.2(h))

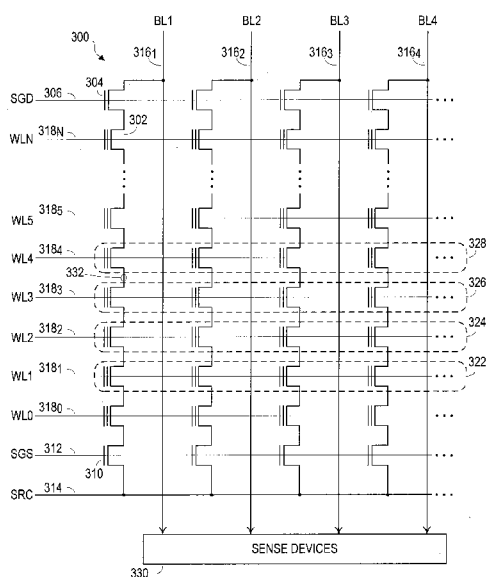
(54) **Title:** MEMORY DEVICES AND OPERATING METHODS FOR A MEMORY DEVICE

FIG. 3

(57) **Abstract:** Devices and methods facilitate memory device operation in all bit line architecture memory devices. In at least one embodiment, memory cells comprising alternating rows are concurrently programmed by row and concurrently sensed by row at a first density whereas memory cells comprising different alternating rows are concurrently programmed by row and concurrently sensed by row at a second density. In at least one additional embodiment, memory cells comprising alternating tiers of memory cells are programmed and sensed by tier at a first density and memory cells comprising different alternating tiers of memory cells are programmed and sensed by tier at a second density.

MEMORY DEVICES AND OPERATING METHODS FOR A MEMORY DEVICE

TECHNICAL FIELD

[0001] The present disclosure relates generally to semiconductor memories and, in particular, in one or more embodiments, the present disclosure relates to operating all bit line memory devices.

BACKGROUND

[0002] Memory devices are typically provided as internal, semiconductor, integrated circuits in computers or other electronic devices. There are many different types of memory including random-access memory (RAM), read only memory (ROM), dynamic random access memory (DRAM), synchronous dynamic random access memory (SDRAM), and flash memory.

[0003] Flash memory devices have developed into a popular source of non-volatile memory for a wide range of electronic applications. Non-volatile memory is memory that can retain its stored data for some extended period without the application of power. Common uses for flash memory and other non-volatile memory include personal computers, personal digital assistants (PDAs), digital cameras, digital media players, digital recorders, games, appliances, vehicles, wireless devices, mobile telephones and removable memory modules, and the uses for non-volatile memory continue to expand.

[0004] Flash memory typically utilizes one of two basic architectures known as NOR Flash and NAND Flash. For example, a NAND flash memory device is a common type of flash memory device, so called for the logical form in which the basic memory cell configuration is arranged and accessed. Typically, the array of memory cells for NAND flash memory devices is arranged such that memory cells are coupled together in series (e.g., coupled source to drain) to form strings of memory cells. Changes in threshold voltage of the memory cells, through programming (which is sometimes referred to as writing) of charge storage structures (e.g., floating gates or charge traps) or other physical phenomena (e.g., phase change or polarization), determine the data value of each cell.

[0005] Two examples of architectures for programming (e.g., writing) and/or sensing (e.g., reading) memory cells of an array are shielded data line (e.g., shielded bit line) and non-shielded data line (e.g., non-shielded bit line) architectures. In a shielded bit line architecture,

memory cells coupled to odd numbered (e.g., alternating) bit lines may be programmed while the adjacent even numbered (e.g., alternating) bit lines are inhibited. During a subsequent programming operation, the odd numbered bit lines may be inhibited while memory cells coupled to the even numbered bit lines are programmed. Thus, the inhibited bit lines facilitate a shielding effect on the strings of memory cells which are being programmed. Sense operations in shielded bit line architecture devices are also performed by sensing alternating selected bit lines while inhibiting unselected alternating bit lines. In non-shielded bit line architectures, commonly referred to as all bit line (ABL) architectures, memory cells coupled to all bit lines may be programmed and sensed concurrently.

[0006] One technique used to increase the memory capacity of a NAND memory device is to form the memory array in a three dimensional (3D) manner. In other words, instead of the series memory strings being formed horizontally on a memory die which is typically referred to as 2D memory, the series strings are formed vertically on a substrate.

[0007] Memory cells, such as flash memory cells, may be configured as what are known in the art as Single Level Memory Cells (SLC) or Multilevel Memory Cells (MLC). SLC and MLC memory cells assign a data state (e.g., representing a respective value of one or more bits) to a specific range of threshold voltages (V_t) stored on the memory cells. Single level memory cells (SLC) permit the storage of a single binary digit (e.g., bit) of data on each memory cell. Meanwhile, MLC technology permits the storage of two or more binary digits per cell (e.g., 2, 3, 4, 5 bits), depending on the quantity of V_t ranges assigned to the cell and the stability of the assigned V_t ranges during the lifetime operation of the memory cell. By way of example, one bit (e.g., 1 or 0) may be represented by two V_t ranges, two bits by four ranges, three bits by eight ranges, etc. Memory cells programmed to store two bits might be referred to as MLC(four-level), as the two bits might be represented by four V_t ranges which might be stored in the memory cells. Memory cells programmed to store three bits might be referred to as MLC(eight-level), as three bits might be represented by eight V_t ranges which might be stored in the memory cells, and so on.

[0008] While memory cells programmed as MLC memory cells are able to store more bits of data per cell than SLC memory cells, MLC memory cells typically have lower reliability characteristics than SLC memory. ABL memory devices are typically programmed wherein all of the memory cells are programmed as SLC or at the same level of MLC levels, for example. Thus, a trade-off generally exists between storing more data in an array of memory

cells and an expected reliability level associated with the data stored at a particular MLC level in the array of memory cells.

[0009] A typically undesirable effect which can occur when programming memory cells is referred to as program disturb. A programming operation performed on a particular memory cell might disturb a programmed data state of one or more nearby (e.g., adjacent) memory cells. This can corrupt data stored in these nearby memory cells. These disturb effects tend to increase as the MLC level increases, such as from programming memory cells as MLC(two-level) to programming memory cells as MLC(three-level), for example.

[0010] For the reasons stated above, and for other reasons stated below which will become apparent to those skilled in the art upon reading and understanding the present specification, there is a need in the art for alternative methods of operating all bit line memory devices.

BRIEF DESCRIPTION OF THE DRAWINGS

[0011] Figure 1 illustrates a schematic representation of an array of NAND configured memory cells.

[0012] Figure 2 illustrates a graphical representation of threshold voltage ranges in a population of memory cells.

[0013] Figure 3 illustrates a schematic representation of a 2D array of NAND configured memory cells.

[0014] Figure 4 illustrates a schematic representation of a portion of a 3D array of NAND configured memory cells.

[0015] Figure 5 illustrates an illustration of a portion of a 3D array of NAND configured memory cells.

[0016] Figure 6 illustrates a graphical representation of a single tier of memory cells.

[0017] Figure 7 is a simplified block diagram of a memory device coupled to a memory access device as part of an electronic system according to an embodiment of the present disclosure.

DETAILED DESCRIPTION

[0018] In the following detailed description of the invention, reference is made to the accompanying drawings that form a part hereof, and in which is shown, by way of illustration, specific embodiments. In the drawings, like numerals describe substantially similar components throughout the several views. Other embodiments may be utilized and structural, logical, and electrical changes may be made without departing from the scope of the present disclosure. The following detailed description is, therefore, not to be taken in a limiting sense.

[0019] Figure 1 illustrates a schematic representation of a shielded bit line NAND architecture memory array 100 wherein charge storage memory cells 102 of the memory array 100 are logically arranged in an array of rows and columns. In a conventional NAND architecture, “rows” refers to two or more memory cells having commonly coupled control gates, while “columns” refers to memory cells coupled as one or more NAND strings of memory cells 102, for example. The schematic representation shown might be representative of a portion of a 2D memory array architecture. The memory cells 102 of the memory array 100 are arranged together in strings (e.g., NAND strings), typically of 8, 16, 32, or more each. Memory cells of a string 102 are connected together in series, source to drain, between a source line 114 and a data line 116, often referred to as a bit line. Each string of memory cells 102 is coupled to source line 114 by a source select gate, such as select gates 110 and to an individual bit line 116 by drain select gates 104, for example. The source select gates 110 are controlled by a source select gate (SGS) control line 112 coupled to their control gates. The drain select gates 104 are controlled by a drain select gate (SGD) control line 106. The one or more strings of memory cells 102 of a memory array 100 are typically arranged in groups (e.g., blocks) of memory cells.

[0020] The memory array 100 is accessed by a string driver (not shown) configured to activate a row of memory cells by selecting a particular access line 118, often referred to as a word line. Each word line 118 (e.g., $WLN-WL_0$ 118_{N-0}) is coupled to a row of memory cells, such as row 120, for example. Bit lines BL1-BL4 116₁-116₄ can be biased to a particular potential depending on the type of operation being performed on the array. Bit lines BL1-BL4 116 are coupled to sense devices (e.g., sense amplifiers) 130 that detect the data state of each cell by sensing voltage or current on a particular bit line 116. As is known

to those skilled in the art, the number of word lines and/or bit lines might be much greater than those shown in Figure 1.

[0021] Programming typically involves applying one or more programming pulses (V_{pgm}) to a selected word line 118, such as WL4 118₄, and thus to the control gates of the row of memory cells 120 coupled to the selected word line 118₄. Typical programming pulses (V_{pgm}) may start at or near 15V and tend to increase in magnitude during each subsequent programming pulse application. While the program potential (e.g., programming pulse) is applied to the selected word line 118₄, a potential, such as a ground potential (e.g., 0V), may be applied to the substrate, and thus to the channels of these memory cells, resulting in a charge transfer from the channel to the charge storage structures of memory cells targeted for programming. For example, floating gates are typically charged through direct injection or Fowler-Nordheim tunneling of electrons from the channel to the floating gate, resulting in a V_t typically greater than zero in a programmed state. In the example of Figure 1, a potential is applied to one or more unselected word lines 118_{N-5} and 118₃₋₀. This potential might be 10V, for example. The potential applied to each unselected word line might be different potentials. A word line adjacent to the selected word line might be biased to a potential of 8V and the next adjacent word line might be biased to 7V, for example. These potentials are not high enough to cause programming of the unselected memory cells. One or more additional unselected word lines might be biased to a potential such as to 0V.

[0022] An inhibit potential is typically applied to bit lines 116 (e.g., V_{cc}) which are not coupled to a NAND string containing a memory cell 102 that is targeted for programming. During a programming operation in a shielded bit line architecture, for example, alternate bit lines 116 may be enabled and inhibited from programming. Even numbered bit lines 116 (e.g., 116₂, and 116₄) might be enabled for programming of memory cells coupled to even numbered bit lines 116_{2,4} while the odd numbered bit lines 116 (e.g., 116₁ and 116₃) are inhibited from programming memory cells coupled to the odd numbered bit lines. A subsequent programming operation might then inhibit the even numbered bit lines 116_{2,4} and enable the odd numbered bit lines 116_{1,3}. For example, the memory cells 102 of row 120 having solid line circles are selected for programming whereas the memory cells 102 having dashed line circles are inhibited from programming as shown in Figure 1.

[0023] Between the application of one or more programming (e.g., V_{pgm}) pulses, a verify operation is typically performed to check each selected memory cell to determine if it has

reached its intended programmed state. If a selected memory cell has reached its intended programmed state it is inhibited from further programming if there remain other memory cells of the selected row still requiring additional programming pulses to reach their intended programmed states. Following a verify operation, an additional programming pulse V_{pgm} is applied if there are memory cells that have not completed programming. This process of applying a programming pulse followed by performing a verify operation typically continues until all the selected memory cells have reached their intended programmed states. If a particular number of programming pulses (e.g., maximum number) have been applied and one or more selected memory cells still have not completed programming, those memory cells might be marked as defective, for example.

[0024] Figure 2 illustrates an example of V_t ranges 200 for a population of MLC (four level) (e.g., 2-bit) memory cells. For example, a memory cell might be programmed to a V_t that falls within one of four different V_t ranges 202-208 of 200 mV, each being used to represent a data state corresponding to a bit pattern comprised of two bits. Typically, a dead space 210 (e.g., sometimes referred to as a margin and might have a range of 200 mV to 400 mV) is maintained between each range 202-208 to keep the ranges from overlapping. As an example, if the V_t of a memory cell is within the first of the four V_t ranges 202, the cell in this case is storing a logical '11' state and is typically considered the erased state of the cell. If the V_t is within the second of the four V_t ranges 204, the cell in this case is storing a logical '10' state. A V_t in the third V_t range 206 of the four V_t ranges would indicate that the cell in this case is storing a logical '00' state. Finally, a V_t residing in the fourth V_t range 208 indicates that a logical '01' state is stored in the cell. The number of levels of programming which might be realized in memory cells (e.g., the potential number of data states to which the cells may be programmed) is sometimes referred to as 'density.' Memory cells programmed as MLC(eight-level) memory cells might be considered to be programmable to a higher density than memory cells which are programmable as MLC(four-level) memory cells, for example.

[0025] A method of programming memory cells of an array of memory cells according to various embodiments of the present disclosure might be described by reference to Figure 3. Figure 3 illustrates a schematic representation of a 2D (e.g., planar) All Bit Line (ABL) NAND architecture memory array 300. In an ABL architecture memory device such as shown in Figure 3, memory cells 302 of the memory array 300 are logically arranged in an

array of rows and columns. Each column of memory cells 302 is coupled between a drain select gate 304 and a source select gate 310. Each row (e.g., 328) of memory cells 302 is coupled to a word line 318. As depicted in Figure 3, each memory cell 302 of a row (e.g., 328) of memory cells 302 shares a source/drain region 332 with a corresponding memory cell 302 of an adjacent row (e.g., 326) of memory cells 302.

[0026] During a programming operation performed on selected memory cells of an ABL architecture memory device, such as represented by Figure 3, each of the bit lines 316 coupled to memory cells of the selected row can be concurrently selected to facilitate performing a program operation on all of the memory cells of the selected row. Concurrently programming memory cells means that the programming of the selected row of memory cells occurs simultaneously for at least a portion of the program operation. For example, programming of those memory cells may be initiated for all of those memory cells at the same time, but one or more of the memory cells may complete their programming prior to other memory cells. Following a programming operation, each of the memory cells of the selected row might be concurrently sensed (e.g., read) to determine if the selected memory cells have reached their intended program states. This is in contrast with an alternating bit line (e.g., shielded bit line) architecture memory device wherein alternating bit lines are selected and memory cells coupled to selected bit lines of a selected row are programmed during a first portion of the programming operation. Subsequently, the remaining memory cells of the selected row in the alternating bit line memory device are programmed during a second portion of the programming operation of a selected row.

[0027] Referring again to Figure 3, the memory cells of a first selected row of memory cells, such as memory cells of row 322, might be concurrently programmed as single level memory cells according to various embodiments of the present description. The memory cells of a second selected row of memory cells, such as memory cells of row 324, might be concurrently programmed as multi-level memory cells during a subsequent programming operation, such as MLC (four-level), for example. Memory cells of a third selected row of memory cells, such as memory cells of row 326, might be concurrently programmed as single level memory cells. The memory cells of a fourth selected row of memory cells, such as memory cells of row 328, might be concurrently programmed as multi-level memory cells, such as MLC (four-level), for example. Thus, according to various embodiments of the present disclosure, memory cells comprising alternating rows of memory cells might be

programmed as SLC memory cells. During subsequent programming operations, the memory cells of each remaining row (e.g., alternate rows) of memory cells might be concurrently programmed as MLC memory cells. In an ABL architecture memory array, each bit line of a plurality of bit lines are concurrently configured for programming during a programming operation performed on each row of memory cells of the array. It should be noted that rows of memory cells, such as rows 322-328, might be programmed in an order other than discussed above according to one or more embodiments of the present disclosure.

[0028] Methods according to various embodiments of the present disclosure are not limited to programming a first number of rows (e.g., alternating rows) of memory cells as SLC memory and a second number of alternating rows of memory cells as MLC(two-level) memory as discussed above. Each row of the array might be programmed to different densities, for example. Still further embodiments might comprise programming rows to one of a plurality of densities wherein each row is separated from another row by rows of memory cells programmed to each other density of the plurality of densities. For example, rows of memory cells might be programmed to one of three densities where each row programmed to a first density is separated by a row of memory cells programmed to a second density and a row of memory cells programmed to a third density. Thus, methods according to one or more embodiments might facilitate programming three or more groups of rows of memory cells each to different densities (e.g., different levels), for example.

[0029] As discussed above, various embodiments are not limited to programming only alternating rows of memory cells as SLC memory and MLC memory. For example, a first number of rows (e.g., alternating rows) of memory cells might be programmed to a first density (e.g., MLC(four-level)) and a second number of rows (e.g., alternating rows) of memory cells might be programmed to a second density (e.g., MLC(eight-level)) memory.

[0030] As discussed above, various memory architectures (e.g., non-volatile memory architectures) are utilized to increase the memory capacity of memory devices. One such architecture is referred to as three-dimensional (3D) memory which incorporates vertical structures which may include semiconductor pillars where at least a portion of each pillar acts as a channel region of the memory cells. Figure 4 illustrates a schematic representation of a 3D all bit line (ABL) architecture NAND memory array 400. The NAND strings of memory cells 402 are each coupled to a bit line BL1-BLN 410 by a drain select gate 404 and to a source SRC 420 by a source select gate 408. Multiple strings of memory cells might be

coupled to the same bit line. Individual memory cell strings can be selected by biasing the SGD lines 412 coupled to activate particular drain select gates 404 between each string of memory cells 402 and a bit line 410. The source select gates 408 can be activated by biasing the SGS signal line 414.

[0031] Each word line 418 is coupled to multiple rows of memory cells 424 of the 3D array. Rows which are commonly coupled to each other by a particular word line, such as rows 424₁-424₃ coupled by word line 418, comprise what are referred to as tiers. However, various embodiments according to the present disclosure are not limited to tiers comprising three rows of memory cells, such as rows 424₁-424₃ shown in Figure 4, for example. Thus, a tier according to one or more embodiments can comprise a plurality of rows of memory cells coupled by a particular word line. 3D arrays, such as illustrated by Figure 4, can comprise multiple tiers of memory cells, for example.

[0032] Figure 5 illustrates a representation of a portion of a 3D array of memory cells 500, such as a portion of a 3D ABL architecture NAND memory device, for example. The figure has been simplified to improve readability. Each cylinder 502 shown in Figure 5 is representative of a memory cell of the 3D array of memory cells 500, for example. Strings of memory cells are made up of vertically formed stacks (e.g., columns) 510 of memory cells. For example, memory cells 502, 504, 506 and 508 might comprise a vertically formed string of memory cells such as the string of memory cells 402 shown in Figure 4, for example. Each column of memory cells 510 is coupled between a source and a respective bit line (not shown in Figure 5), such as between source SRC 420 and bit lines 410, respectively, as shown in Figure 4.

[0033] Figure 5 further illustrates that the 3D array of memory cells 500 is further arranged in a number of tiers 512-518 of memory cells. For example, the uppermost tier 512 of memory cells of the array 500 might comprise a first tier of memory cells. The tier 514 of memory cells below tier 512 might comprise a second tier of memory cells, and so on. Figure 6 illustrates a single tier (e.g., plane) 600 of memory cells 602. Thus, the 3D ABL architecture array of memory cells 500 shown in Figure 5 might comprise a stack of four planes of memory cells 600 such as shown in Figure 6. Each column of memory cells 510 comprises a memory cell 502-508 from each tier 512-518 of memory cells of the 3D memory array 500.

[0034] Each tier of memory cells 512-518 has a particular word line (not shown in Figure 5) associated with it. For example, the memory cells comprising the uppermost tier 512 of memory cells might be coupled to a single word line, such as word line 418 shown in Figure 4, for example. The remaining tiers 514-518 of memory cells shown in Figure 5 are each coupled to their own respective word lines (not shown in Figure 5). Thus, in a 3D ABL architecture memory array, such as illustrated in Figure 5, each memory cell of a particular tier of memory cells might be programmed by biasing the associated word line and each of the plurality of bit lines coupled to the strings of memory cells comprising the particular tier of memory cells. According to various embodiments, a first tier of memory cells might be programmed as SLC memory, a second tier might be programmed as MLC memory, a third tier might be programmed as SLC memory, and so on.

[0035] A programming method according to one or more embodiments of the present disclosure can be discussed by way of example with reference to Figure 5. Memory cells of the first tier of memory cells 512 might be selected for programming during a first programming operation. As the word line coupled to the first tier is coupled to each memory cell of the first tier, each memory cell of the first tier will be biased to one or more programming pulses applied to the first word line. Each bit line of the array coupled to each memory cell of the first tier of memory cells is also configured to allow concurrent programming of memory cells of the first tier. Word lines coupled to remaining tiers of memory cells might be biased such as to mitigate programming (e.g., further programming) of memory cells comprising the remaining tiers of memory cells.

[0036] Tiers of memory cells might also be sensed, such as part of a memory device read operation, at the particular density to which each tier of memory cells has been programmed to. For example, tiers of memory cells programmed as SLC memory cells can be sensed (e.g., read) as SLC memory cells. Tiers of memory cells programmed as MLC memory cells can be sensed (e.g., read) as MLC memory cells, etc. Word lines coupled to tiers of unselected memory cells might be biased to a Vpass potential to activate the unselected memory cells to operate in a pass through mode regardless of their respective programmed states while the selected memory cells are sensed.

[0037] Thus, according to various embodiments of the present disclosure, alternating tiers of memory cells in a 3D ABL architecture NAND memory device might be programmed to a first program density and different alternating tiers of memory cells might be programmed to

a second program density. For example, a particular tier of memory cells might be programmed as MLC memory cells. A tier above and below (e.g., directly above and below) the particular tier of memory cells might be programmed as SLC memory. This facilitates tailoring a trade-off between capacity and reliability in programming the memory array. For example, an average of 1.5 bits per memory cell might be realized by programming half of the tiers as SLC memory cells and programming the other half of tiers as MLC(four-level) (e.g., MLC(two-bit)) memory cells, for example. Different combinations of memory cell program densities might be utilized according to various embodiments of the present disclosure. Each tier of tiers 512-518 might be programmed to a different density, for example.

[0038] Figure 7 is a functional block diagram of an electronic system having at least one apparatus, such as a memory device 700, according to one or more embodiments of the present disclosure. As used herein, an ‘apparatus’ can refer to, but is not limited to, an array, circuitry, a device or devices, a semiconductor die or dies, a module or modules and/or a system or systems, for example. The memory device 700 illustrated in Figure 7 is coupled to a memory access device, such as a processor 710. The processor 710 may be a microprocessor or some other type of controlling circuitry. The memory device 700 and the processor 710 form part of an electronic system 720. The memory device 700 has been simplified to focus on features of the memory device that are helpful in understanding various embodiments of the present disclosure.

[0039] The memory device 700 includes one or more memory arrays 730 that might be logically arranged in rows and columns and may be further configured as all bit line (ABL) memory arrays. The one or more memory arrays 730 might comprise 2D and/or 3D memory arrays. According to various embodiments of the present disclosure, the memory cells of memory array 730 are flash memory cells configured as a 2D ABL architecture NAND memory array. According to one or more embodiments, the memory cells of memory array 730 are flash memory cells configured as a 3D ABL architecture NAND memory array. The memory array 730 might include multiple banks and blocks of memory cells residing on a single or multiple die as part of the memory device 700. Memory array 730 might comprise SLC and/or MLC memory. The memory array 730 might also be adaptable to store varying densities (e.g., MLC(four-level) and MLC(eight-level)) of data in each memory cell, for example.

[0040] An address buffer circuit 740 is provided to latch address signals provided on address input connections A0-Ax 742. Address signals are received and decoded by a row decoder 744 and a column decoder 748 to access the memory array 730. Row decoder 744 might comprise driver circuitry configured to bias the word lines of the memory array 730, for example. It will be appreciated by those skilled in the art, with the benefit of the present description, that the number of address input connections 742 might depend on the architecture of the memory array 730. That is, the number of address digits increase with both increased memory cell counts and increased bank and block counts, for example.

[0041] The memory device 700 reads data in the memory array 730 by sensing voltage or current changes in the memory array columns using sense devices, such as sense/data cache circuitry 750. The sense/data cache circuitry 750, in at least one embodiment, is coupled to read and latch a row of data from the memory array 730. The sense/data cache circuitry 750 might comprise driver circuitry to bias bit lines to various potentials according to one or more embodiments of the present disclosure. Data input and output (I/O) buffer circuitry 760 is included for bi-directional data communication over a plurality of data connections 762 with the processor 710. Write/erase circuitry 756 is provided to write data to or to erase data from the memory array 730.

[0042] Control circuitry 770 is configured, at least in part, to facilitate implementing various embodiments of the present disclosure. Control circuitry 770 might be coupled (not shown in Figure 7) to one or more of the elements of the memory device 700. For example, the control circuitry might be coupled to the row decoder 744 and configured to cause the row decoder driver circuitry to bias particular word lines of the memory array 730 according to various embodiments of the present disclosure. Control circuitry 770 might be coupled (not shown in Figure 7) to and configured to cause the sense/data cache driver circuitry 750 to bias particular bit lines of the array 730 according to one or more embodiments. For example, the control circuitry might be configured to operate the memory array 730 as an ABL memory array. Control circuitry 770 might further comprise one or more register circuits. Programming these registers facilitates configuring the memory device to program and/or sense the memory cells in the memory array according to various embodiments as discussed above. For example, one or more registers might be programmed to indicate (e.g., assign) which rows and/or tiers of memory cells are to be programmed as SLC or a particular level of MLC memory cells such as described above. In one or more embodiments, control

circuitry 770 and/or firmware or other circuitry can individually, in combination, or in combination with other elements, form an internal controller. As used herein, however, a controller need not necessarily include any or all of such components. In some embodiments, a controller can comprise an internal controller (e.g., located on the same die as the memory array) and/or an external controller. In at least one embodiment, the control circuitry 770 may utilize a state machine.

[0043] Control signals and commands can be sent by the processor 710 to the memory device 700 over the command bus 772. The command bus 772 may be a discrete signal or may be comprised of multiple signals, for example. These command signals 772 are used to control the operations on the memory array 730, including data read, data write (e.g., program), and erase operations. The command bus 772, address bus 742 and data bus 762 may all be combined or may be combined in part to form a number of standard interfaces (e.g., communications interfaces) 778. For example, the interface 778 between the memory device 700 and the processor 710 might be a Universal Serial Bus (USB) interface. The interface 778 might also be a standard interface used with many hard disk drives (e.g., SATA, PATA) as are known to those skilled in the art.

[0044] The electronic system illustrated in Figure 7 has been simplified to facilitate a basic understanding of the features of the memory and is for purposes of illustration only. A more detailed understanding of internal circuitry and functions of non-volatile memories are known to those skilled in the art.

CONCLUSION

[0045] In summary, one or more embodiments of the present disclosure provide methods of programming and sensing memory cells in memory arrays, such as 2D ABL and/or 3D ABL architecture NAND memory arrays. The programming and/or sensing operations discussed above according to various embodiments might be performed on a number of memory array configurations such as the 3D ABL architecture NAND memory arrays as well as 2D ABL architecture NAND memory arrays. Memory cells comprising alternating rows and/or tiers of memory cells might be programmed and/or sensed at different program densities than other rows and/or tiers of memory cells of a memory array. These methods might facilitate selecting a desired level of expected reliability in the memory array while facilitating an

increase in data storage density in the memory array. A reduction in program disturb might also be realized.

[0046] Although specific embodiments have been illustrated and described herein, it will be appreciated by those of ordinary skill in the art that other configurations may be substituted for the specific embodiments shown. Many adaptations of the disclosure will be apparent to those of ordinary skill in the art. Accordingly, this application is intended to cover any adaptations or variations of the disclosure.

What is claimed is:

1. A method of operating an apparatus having an array of memory cells logically arranged in rows and in columns, the method comprising:
concurrently programming each memory cell of a first row of memory cells at a first density; and
concurrently programming each memory cell of a second row of memory cells at a second density different than the first density.
2. The method of claim 1, wherein:
concurrently programming each memory cell of a first row of memory cells at a first density comprises concurrently programming each memory cell of the first row of memory cells to a respective one of a first number of data states; and
concurrently programming each memory cell of a second row of memory cells at a second density different than the first density comprises concurrently programming each memory cell of the second row of memory cells to a respective one of a second number of data states different than the first number of data states.
3. The method of claim 2, wherein each column of the array of memory cells is coupled to a respective one of a plurality of data lines, and wherein at least one column of the array of memory cells comprises a memory cell of the first row of memory cells and a memory cell of the second row of memory cells.
4. The method of any of claims 1-3, further comprising concurrently programming each memory cell of a third row of memory cells at a third density different than the first density and the second density.
5. The method of any of claims 1-3, further comprising concurrently programming each memory cell of a third row of memory cells at the first density, wherein the first row of memory cells is adjacent to the second row of memory cells and where the second row of memory cells is adjacent to the third row of memory cells.

6. The method of claim 5, further comprising concurrently programming each memory cell of a fourth row of memory cells at the second density, wherein the third row of memory cells is adjacent to the fourth row of memory cells.
7. The method of any of claims 1-3, wherein the first row of memory cells comprises two or more adjacent memory cells commonly coupled to a particular access line and the second row of memory cells comprises two or more adjacent memory cells commonly coupled to an access line other than the particular access line.
8. The method of any of claims 1-3, wherein each memory cell of the first row of memory cells shares a source/drain region with a corresponding memory cell of the second row of memory cells.
9. The method of any of claims 1-3, wherein the array of memory cells is further logically arranged in tiers with each tier comprising a plurality of rows of memory cells, and wherein:
concurrently programming each memory cell of a first row of memory cells at a first density comprises concurrently programming each memory cell of each row of memory cells of a first tier of memory cells at the first density; and
concurrently programming each memory cell of a second row of memory cells at a second density different than the first density comprises concurrently programming each memory cell of each row of memory cells of a second tier of memory cells at the second density.
10. The method of claim 9, wherein memory cells of the first tier of memory cells are commonly coupled to a first access line, and wherein memory cells of the second tier of memory cells are commonly coupled to a second access line.
11. The method of claim 10, wherein each memory cell of the first tier of memory cells shares a source/drain region with a corresponding memory cell of the second tier of memory cells.

12. The method of claim 9, further comprising concurrently programming a selected memory cell of each row of memory cells comprising a third tier of memory cells at the first density, wherein the first tier of memory cells is adjacent to the second tier of memory cells and where the second tier of memory cells is adjacent to the third tier of memory cells.
13. The method of claim 12, further comprising concurrently programming a selected memory cell of each row of memory cells comprising a fourth tier of memory cells at the second density, wherein the third tier of memory cells is adjacent to the fourth tier of memory cells.
14. An apparatus, comprising:
an array of memory cells logically arranged in rows and in columns, where each row is coupled to a respective access line and where each column is coupled to a respective data line; and
a controller, wherein the controller is configured to cause each memory cell of a first row of memory cells to be concurrently programmed to a respective one of a first number of data states, and to cause each memory cell of a second row of memory cells to be concurrently programmed to a respective one of a second number of data states;
wherein the first number of data states is different than the second number of data states.
15. The apparatus of claim 14, wherein the controller is further configured to cause each memory cell of a third row of memory cells to be concurrently programmed to a respective one of the first number of data states, where the second row of memory cells is between the first row of memory cells and the third row of memory cells.
16. The apparatus of claim 14, wherein the controller is further configured to cause each memory cell of the first row of memory cells to be concurrently sensed, and to cause each memory cell of the second row of memory cells to be concurrently sensed.

17. The apparatus of any of claims 14-16, wherein each data line comprises a bit line and where the memory array comprises a two-dimensional (2D) all bit line (ABL) architecture memory array.
18. The apparatus of any of claims 14-16, wherein each data line comprises a bit line and where the memory array comprises a three-dimensional (3D) all bit line (ABL) architecture memory array.
19. The apparatus of any of claims 14-16, wherein the array of memory cells is further arranged in a plurality of tiers of memory cells where each tier of memory cells comprises a plurality of rows of memory cells.
20. The apparatus of claim 19, wherein the controller is further configured to cause each memory cell of a first tier of memory cells to be programmed to a respective one of the first number of data states, and to cause each memory cell of a second tier of memory cells to be programmed to a respective one of the second number of data states.
21. The apparatus of claim 19, wherein the controller is further configured to cause each memory cell comprising a first number of alternating tiers of memory cells to be programmed to a respective one of the first number of data states, and to cause each memory cell comprising a second number of alternating tiers of memory cells to be programmed to a respective one of the second number of data states.

1 / 6

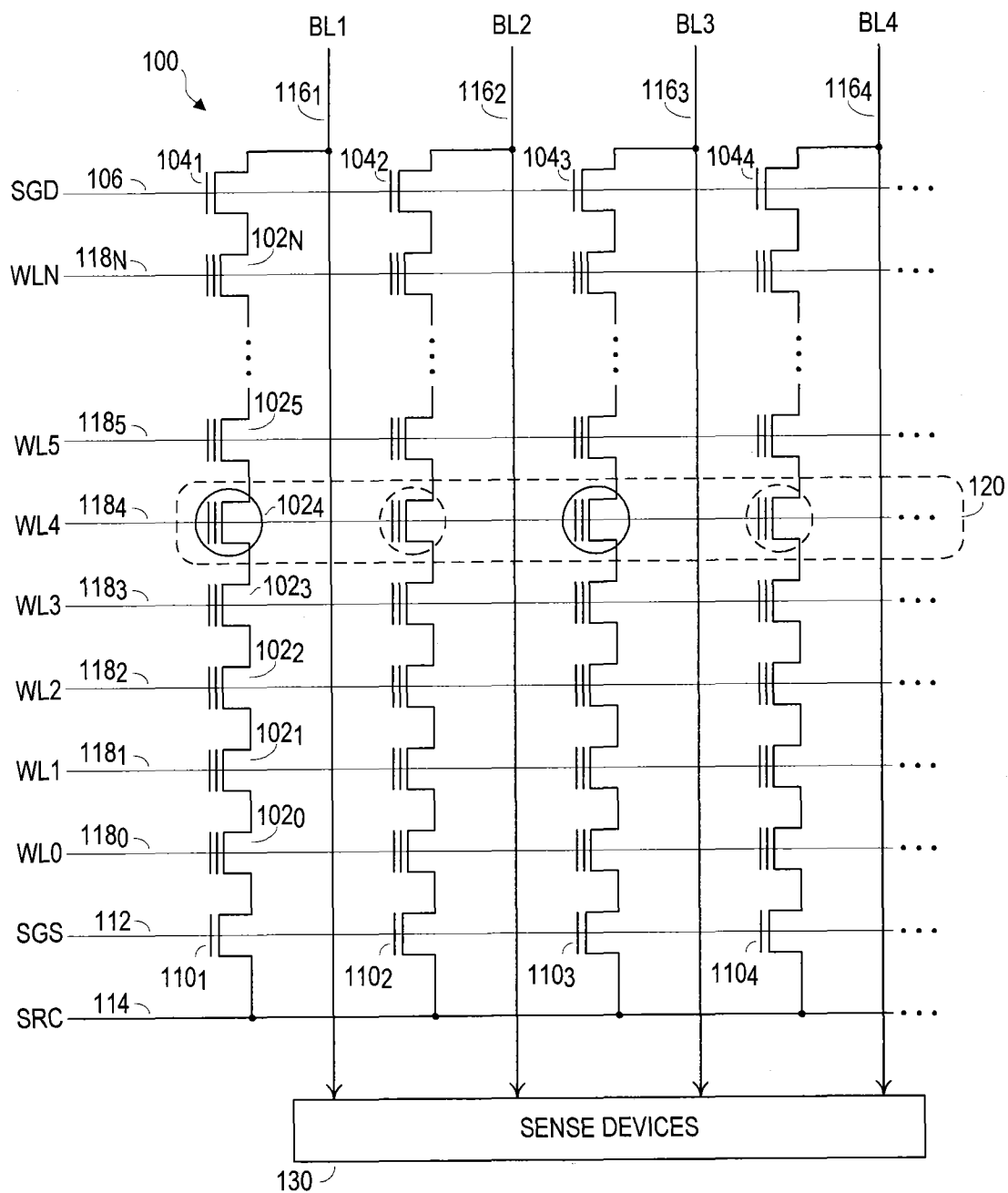


FIG. 1
BACKGROUND ART

2 / 6

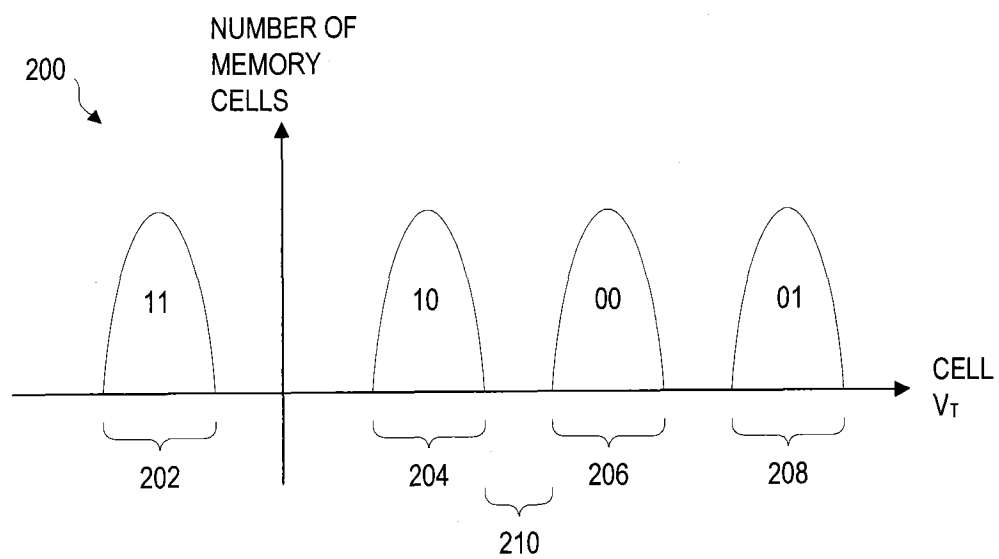


FIG. 2
BACKGROUND ART

3 / 6

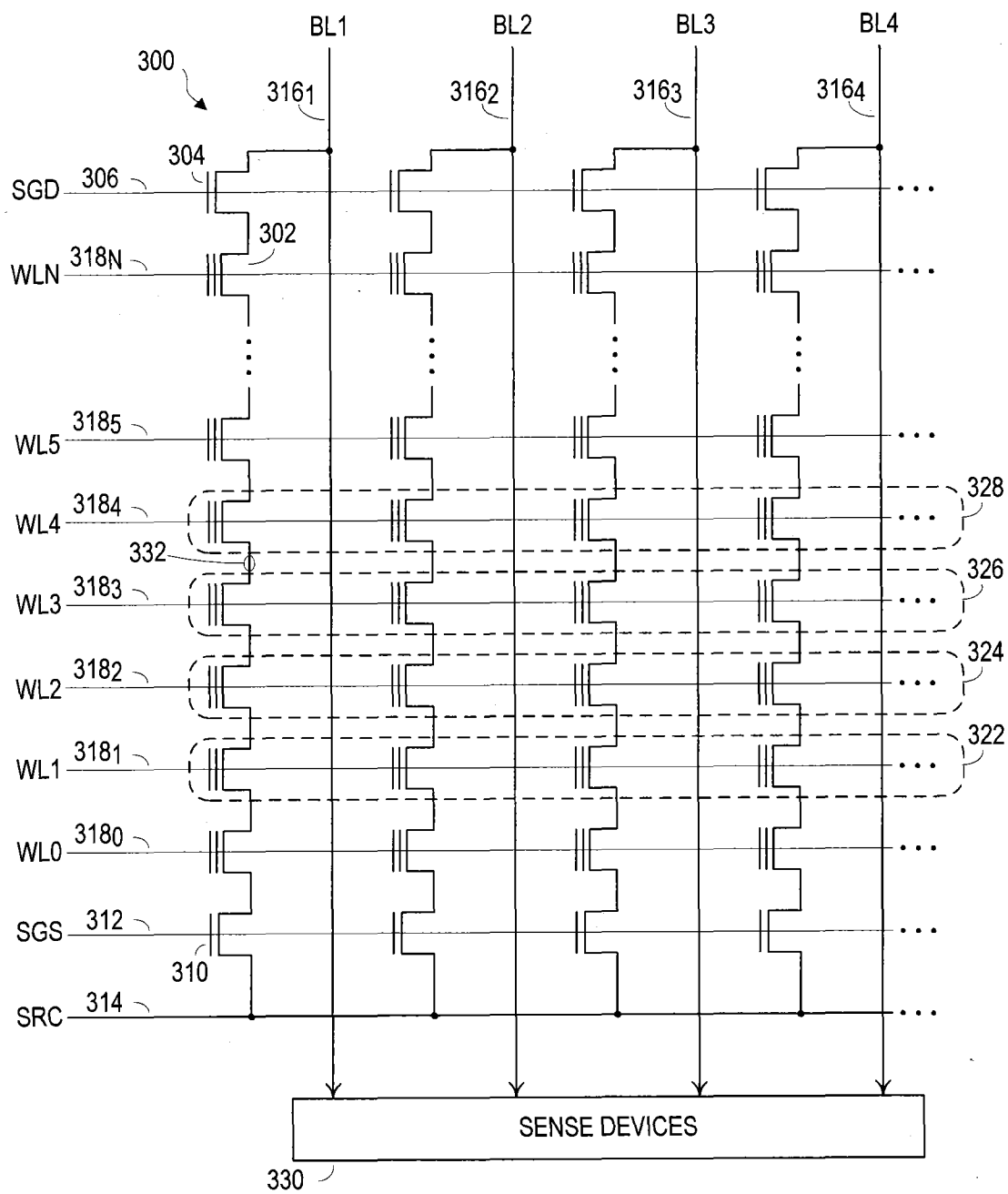


FIG. 3

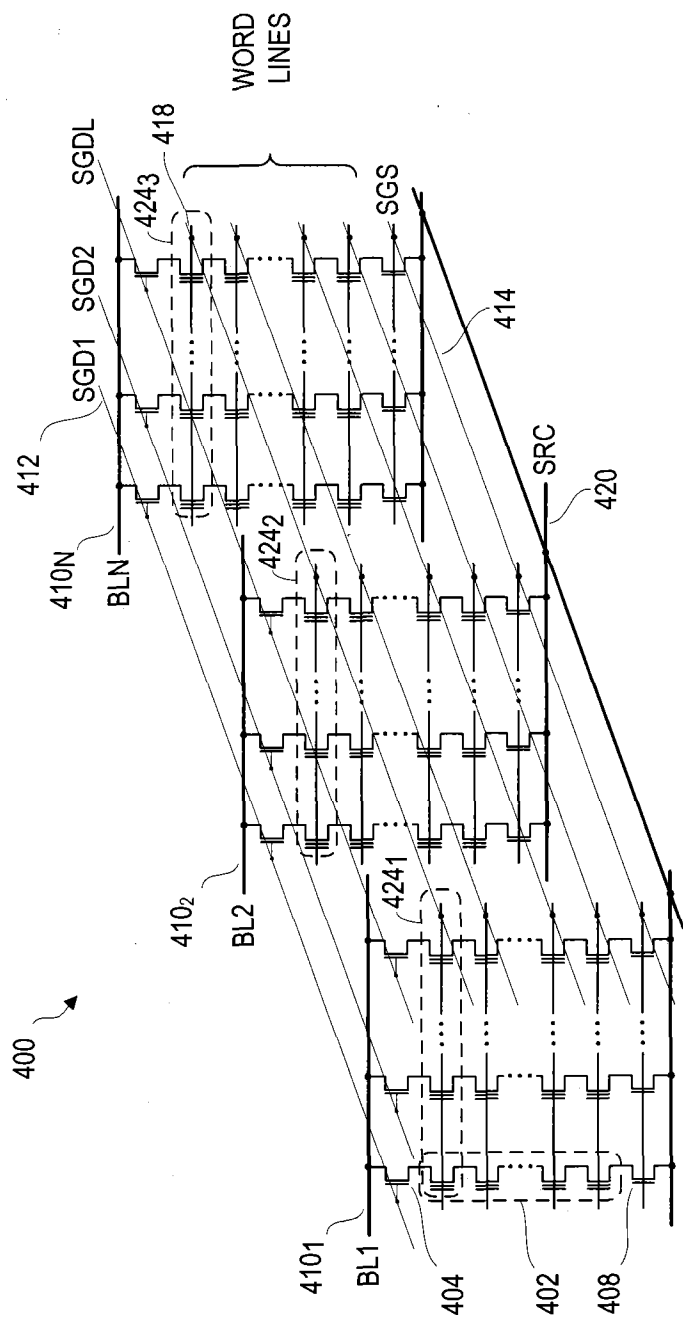


FIG. 4

5 / 6

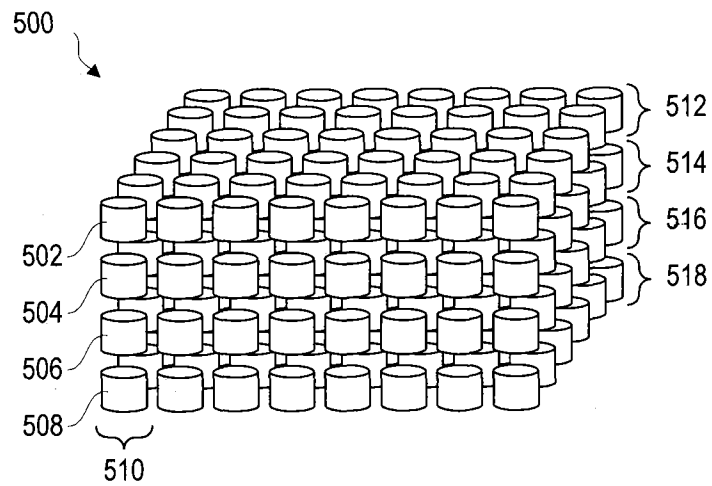


FIG. 5

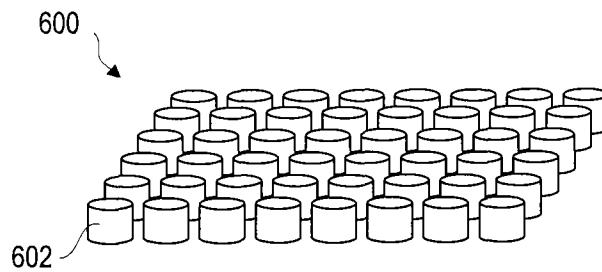


FIG. 6

6 / 6

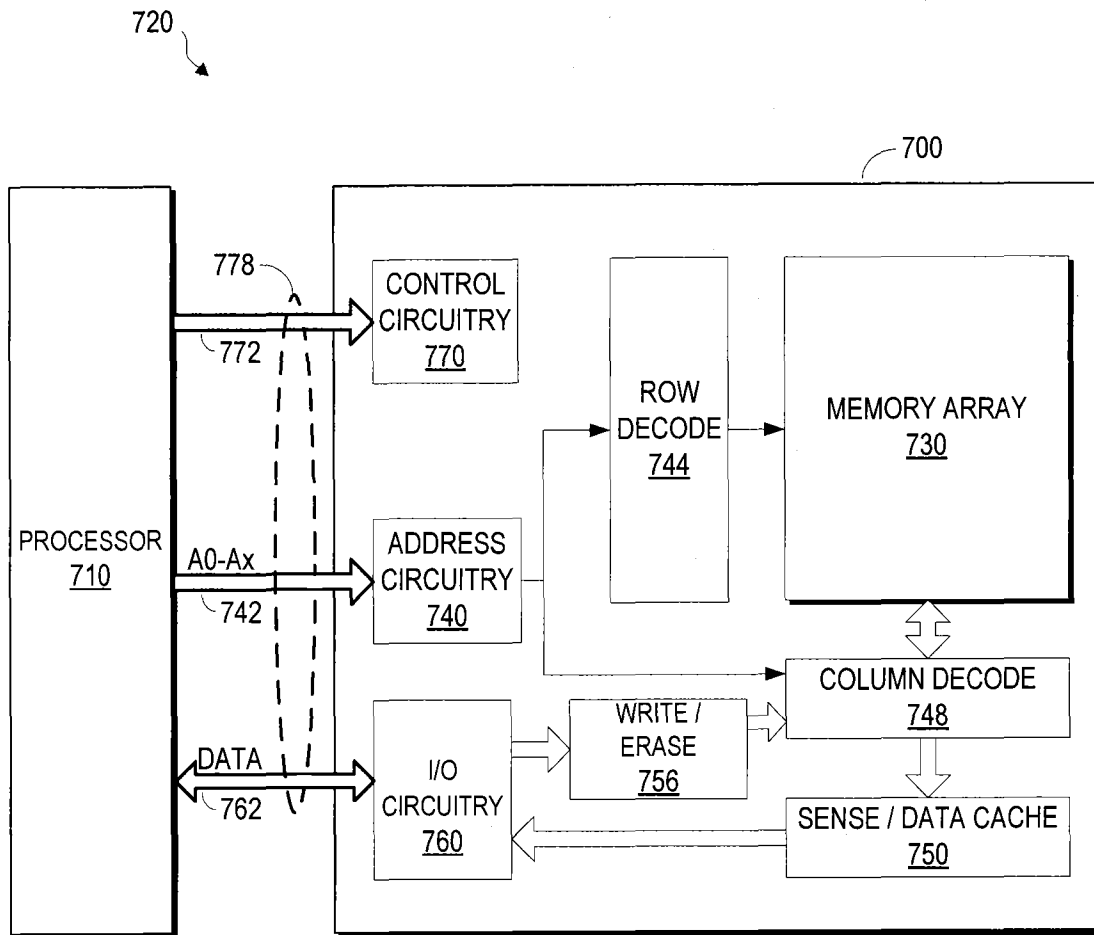


FIG. 7

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2013/056826**A. CLASSIFICATION OF SUBJECT MATTER****G11C 16/10(2006.01)i, G11C 16/04(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 16/10; G11C 8/00; G11C 5/06; G11C 16/04; H03M 13/15; H01L 21/336; G06F 11/10; G11C 16/06

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, cell, bit, density

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2009-0296478 A1 (YOU SUNG KIM et al.) 03 December 2009 See abstract, paragraphs [0007],[0024]-[0034], claims 1-12, and figures 1A-4.	1-8,14-18
Y		9-13,19-21
Y	US 2009-0231942 A1 (MING-HSIU LEE et al.) 17 September 2009 See abstract, paragraph [0007], claims 1-9, and figures 1-4.	9-13,19-21
A	US 2009-0168513 A1 (TOMO HARU TANAKA) 02 July 2009 See abstract, claims 1-25, and figures 3-9.	1-21
A	US 2011-0302475 A1 (ROBERT B. EISENHUTH et al.) 08 December 2011 See abstract, paragraph [0009], and claims 11-14,25-27.	1-21
A	US 2006-0172493 A1 (KIRK D. PRALL) 03 August 2006 See abstract, claims 1-40, and figure 1.	1-21



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

30 December 2013 (30.12.2013)

Date of mailing of the international search report

30 December 2013 (30.12.2013)

Name and mailing address of the ISA/KR

Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon Metropolitan City,
302-701, Republic of Korea

Facsimile No. +82-42-472-7140

Authorized officer

HWANG, Yun Koo

Telephone No. +82-42-481-5715



INTERNATIONAL SEARCH REPORT

International application No.

PCT/US2013/056826

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2009-0296478 A1	03/12/2009	KR 10-0967001 B1 KR 2009-0124096 A US 8050103 B2	30/06/2010 03/12/2009 01/11/2011
US 2009-0231942 A1	17/09/2009	CN 100505268 C CN 1841748 A0 TW I317950 A TW I317950 B US 2006-0208304 A1 US 2006-0233082 A1 US 2011-0121411 A1 US 7554873 B2 US 7589368 B2 US 7911856 B2	24/06/2009 04/10/2006 01/12/2009 01/12/2009 21/09/2006 19/10/2006 26/05/2011 30/06/2009 15/09/2009 22/03/2011
US 2009-0168513 A1	02/07/2009	JP 04534211 B2 JP 2009-158014 A TW 200935429 A WO 2009-085663 A2 WO 2009-085663 A3	01/09/2010 16/07/2009 16/08/2009 09/07/2009 27/08/2009
US 2011-0302475 A1	08/12/2011	CN 102934171 A EP 2577672 A2 KR 10-2013-0050335 A TW 201214439 A WO 2011-153000 A2 WO 2011-153000 A3	13/02/2013 10/04/2013 15/05/2013 01/04/2012 08/12/2011 23/02/2012
US 2006-0172493 A1	03/08/2006	TW I262590 B US 2005-0184329 A1 US 2006-0163614 A1 US 7112815 B2 US 7157305 B2 WO 2005-083715 A1	21/09/2006 25/08/2005 27/07/2006 26/09/2006 02/01/2007 09/09/2005