



(51) International Patent Classification:
G11C 13/00 (2006.01)

(21) International Application Number:
PCT/US2015/013508

(22) International Filing Date:
29 January 2015 (29.01.2015)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant: HEWLETT PACKARD ENTERPRISE DEVELOPMENT LP [US/US]; 11445 Compaq Center Drive West, Houston, TX 77070 (US).

(72) Inventors: GE, Ning; 1501 Page Mill Rd., Palo Alto, California 94304-1100 (US). YANG, Jianhua; 1501 Page Mill Rd., Palo Alto, California 94304-1100 (US). LI, Zhiyong; 1501 Page Mill Rd., Palo Alto, California 94304-1100 (US). WILLIAMS, Stanley; 1501 Page Mill Rd., Palo Alto, California 94304-1100 (US).

(74) Agents: COLLINS, David W. et al.; Hewlett Packard Enterprise, 3404 E. Harmony Road, Mail Stop 79, Fort Collins, CO 80528 (US).

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

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

Declarations under Rule 4.17:

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

[Continued on next page]

(54) Title: HAMMING DISTANCE COMPUTATION

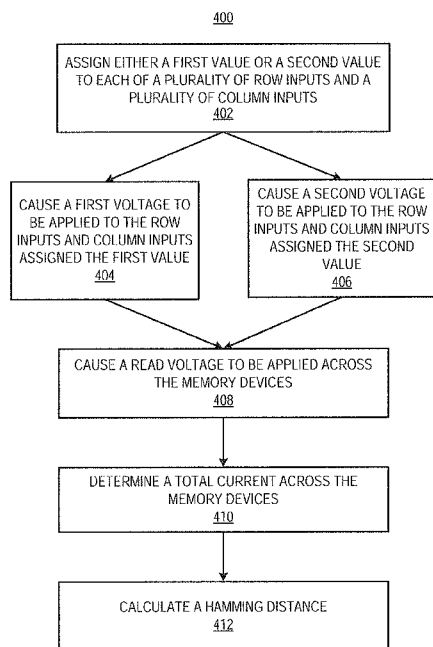


FIG. 4

(57) Abstract: In an example, an apparatus for computing hamming distance may include a memristor array formed of a plurality of memristors, row inputs, and column inputs. The apparatus may also include a processor that is to: assign one of a first value and a second value to each of the row inputs and the column inputs, in which the row inputs correspond to characters in a first character string and the column inputs correspond to characters in a second character string, cause a first voltage to be applied to the row inputs and the column inputs that are assigned the first value and a second voltage to be applied to the row inputs and the column inputs that are assigned the second value; and calculate a hamming distance between the first character string and the second character string based upon the states of the memristors.



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

Published:

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

HAMMING DISTANCE COMPUTATION

BACKGROUND

[0001] Computer system architectures continue to evolve to meet ever-increasing computational requirements. One evolved approach is to use multiple core processors, in which the multiple cores perform tasks in parallel. Another approach is to use accelerators, which are separate architectural substructures that are architected using a different set of special objectives than the base core processor. The special objectives are typically derived from the needs of a special class of applications. Additionally, the accelerators are often integrated with a core or attached to appliances via interconnects.

BRIEF DESCRIPTION OF THE DRAWINGS

[0002] Features of the present disclosure are illustrated by way of example and not limited in the following figure(s), in which like numerals indicate like elements, in which:

[0003] FIG. 1 shows a simplified block diagram of an apparatus for computing hamming distance, according to an example of the present disclosure;

[0004] FIG. 2 shows a simplified isometric view of a portion of the memristor array depicted in FIG. 1, according to an example of the present disclosure;

[0005] FIG. 3 shows a simplified view of an apparatus for computing hamming distance, according to another example of the present disclosure;

[0006] FIGS. 4 and 5, respectively show flow diagrams of methods for computing a hamming distance between a first character string and a second character string, according to two examples of the present disclosure; and

[0007] FIG. 6 shows a schematic representation of a computing device, which may implement the apparatuses depicted in FIGS. 1 and 3, according to an example of the present disclosure.

DETAILED DESCRIPTION

[0008] For simplicity and illustrative purposes, the present disclosure is described by referring mainly to an example thereof. In the following description, numerous specific details are set forth in order to provide a thorough understanding of the present disclosure. It will be readily apparent however, that the present disclosure may be practiced without limitation to these specific details. In other instances, some methods and structures have not been described in detail so as not to unnecessarily obscure the present disclosure. As used herein, the terms “a” and “an” are intended to denote at least one of a particular element, the term “includes” means includes but not limited to, the term “including” means including but not limited to, and the term “based on” means based at least in part on.

[0009] Disclosed herein is an apparatus for computing the hamming distance between a first character string and a second character string through use of a memristor array. That is, in contrast to conventional use of memristor arrays as memory devices to store data, the memristor array disclosed herein may be used as an accelerator of a processor. Particularly, a processor may implement the memristor array disclosed herein to accelerate the computation of the hamming distance between the first character string and the second character string. In one regard, a processor may implement the memristor array disclosed herein to compute the hamming distance between two character strings in a manner that is relatively faster and more efficient than if the processor were to compute the hamming distance between the character strings itself. As such, the memristor array may be implemented as an accelerator of a processor, in which the memristor array may be implemented solely in the computation of a hamming distance between two character strings. By way of example, through implementation of apparatus disclosed herein, the hamming distance between the two character strings may be determined in three clock cycles as compared with a much larger number of clock cycles required if the processor were to determine the hamming distance itself.

[0010] As discussed in greater detail herein, the memristor array may be implemented to compute the hamming distance through a determination of how the current through the memristor array is affected following application of voltages through the memristor array corresponding to the characters in the two character strings. That is, the characters of a first character string may be assigned to the row inputs of the memristor array in the order in which the characters are arranged in the first character string. Likewise, the characters of a second character string may be assigned to the column inputs of the memristor array in the order in which the characters are arranged in the second character string. Voltages corresponding to the characters (or converted values related to the characters) may be applied to each of the row inputs and the column inputs. Differences in voltages across the memristors may cause the resistance states of the memristors to be changed. As such, the hamming distance between the first character string and the second character string may be related to the increase in resistance level through the memristor array resulting from the operation described above.

[0011] With reference first to FIG. 1, there is shown a simplified block diagram of an apparatus 100 for computing hamming distance, according to an example. It should be understood that the apparatus 100 depicted in FIG. 1 may include additional components and that some of the components described herein may be removed and/or modified without departing from a scope of the apparatus 100.

[0012] As shown in FIG. 1, the apparatus 100 includes a substrate 110, a processor 120, and a memristor array 130, which is also referenced herein as a memory device array. According to an example, the substrate 110 is an integrated circuit chip on which the processor 120 and the memristor array 130 are attached. In this example, the processor 120 and the memristor array 130 may communicate directly to each other through a connection on the substrate 110. The processor 120 may be a central processing unit (CPU), a microprocessor, a micro-controller, an application specific integrated circuit (ASIC), a processor core, or the like.

[0013] As discussed herein, the processor 120 may implement the memristor array 130 as an accelerator to accelerate the computation of a hamming distance between two character strings. That is, the processor 120 may implement the memristor array 130 to compute the hamming distance between two character strings in a manner that is relatively faster and more efficient than if the processor 120 were to compute the hamming distance between the character strings itself. In this regard, the memristor array 130 may be implemented as an accelerator of the processor 120, in which the memristor array 130 is implemented solely in the computation of a hamming distance between two character strings.

[0014] The hamming distance between two character strings may be defined as the number of positions at which symbols (or characters) corresponding to the two character strings are different from each other. In other words, the hamming distance between two character strings may be defined as a measure of the minimum number of substitutions required to change one character string into the other. By way of example, the hamming distance between the following character strings "1011101" and "1001001" is 2. As another example, the hamming distance between the following character strings "karolin" and "kathrin" is 3. Hamming distances may be used in any of a number of disciplines including, for instance, information theory, coding theory, cryptography, systematics, etc.

[0015] Although not shown in FIG. 1, the apparatus 100 may include a memory, which may be, for instance, a volatile or non-volatile memory, such as dynamic random access memory (DRAM), electrically erasable programmable read-only memory (EEPROM), magnetoresistive random access memory (MRAM), or the like. In this example, the machine readable instructions, for instance, corresponding to the methods disclosed herein, may be stored in the memory. According to an example, the memory (not shown) is another memristor array on which the processor is to store data. In this example, the memristor array 130, which may operate solely as an accelerator, may be

fabricated with the another memristor array, which may operate as a memory for the processor 120.

[0016] Turning now to FIG. 2, there is shown a simplified isometric view of a portion of the memristor array 130 depicted in FIG. 1, according to an example. It should be understood that the memristor array 130 depicted in FIG. 2 may include additional components and that some of the components described herein may be removed and/or modified without departing from a scope of the memristor array 130.

[0017] As shown in FIG. 2, the memristor array 130 may be formed as a crossbar array and may include a first layer 210 formed of a plurality of first electrodes 212 and a second layer 220 formed of a plurality of second electrodes 222. The first electrodes 212 are depicted as extending along a first plane and the second electrodes 222 are depicted as extending along a second plane, in which the second plane is parallel or nearly parallel to the first plane.

[0018] The first electrodes 212 and the second electrodes 222 are also depicted as being in a crossed relationship with respect to each other such that junctions 230 are formed at intersections between respective pairs of the first electrodes 212 and second electrodes 222. That is, the first electrodes 212 are depicted as extending in a direction that is perpendicular to the direction in which the second electrodes 222 extend. According to an example, the second electrodes 222 may be substantially perpendicular to the first electrodes 212, e.g., there may be less than about a 5° of rotation difference between the first electrodes 212 and the second electrodes 222.

[0019] The second electrodes 222 are further depicted as being in a spaced relationship with respect to the first electrodes 212 such that gaps exists between the first electrodes 212 and the second electrodes 222 into which switching elements 232 may be provided. The switching elements 232 and the sections of the first electrodes 212 and the second electrodes 222 around the switching elements 232 may form respective memristors 240, which are also referred to herein as memory devices.

[0020] The first electrodes 212 may be formed of an electrically conductive material, such as AlCu, AlCuSi, AlCuSi with a barrier layer, such as TiN, or the like. The second electrodes 222 may be formed of any of the example materials listed for the first electrodes 212. In addition, the second electrodes 222 may be formed of the same or different materials as compared with the first electrodes 212. The second electrodes 222 may be formed of an electrically conductive material, such as TaAl, WSiN, AlCu combination, or the like. The switching elements 232 may be formed of switching oxides, such as a metallic oxide. Specific examples of switching oxide materials may include magnesium oxide, titanium oxide, zirconium oxide, hafnium oxide, vanadium oxide, niobium oxide, tantalum oxide, chromium oxide, molybdenum oxide, tungsten oxide, manganese oxide, iron oxide, cobalt oxide, copper oxide, zinc oxide, aluminum oxide, gallium oxide, silicon oxide, germanium oxide, tin dioxide, bismuth oxide, nickel oxide, yttrium oxide, gadolinium oxide, and rhenium oxide, among other oxides. In addition to the binary oxides presented above, the switching oxides may be ternary and complex oxides such as silicon oxynitride. The oxides presented may be formed using any of a number of different processes such as sputtering from an oxide target, reactive sputtering from a metal target, atomic layer deposition (ALD), oxidizing a deposited metal or alloy layer, etc.

[0021] A memristor 240 may generally be defined as an electrically actuated apparatus formed of a pair of spaced apart electrodes 212, 214 with a switching element 232 positioned between the electrodes 212, 214. The memristors 240 are able to change the values of their resistances in response to various programming conditions and are able to exhibit a memory of past electrical conditions. For instance, the memristors 240 may be programmed to respectively represent a logical "1" or ON while in a low resistance state and a logical "0" or OFF while in a high resistance state and may retain these resistance states. Particularly, the resistance state of the switching element 232 may be changed through application of a current, in which the current may cause mobile dopants in the switching element to move, which may alter the

electrical operation of the memristor 240. After removal of the current, the locations and characteristics of the dopants remain stable until the application of another programming electrical field. The states of the memristors 240 may be read by applying a lower reading voltage across the switching element which allows the internal electrical resistance of the memristor 240 to be sensed but does not generate a sufficiently high electrical field to cause significant dopant motion.

[0022] With reference now to FIG. 3, there is shown a simplified view of an apparatus 300 for computing hamming distance, according to another example. It should be understood that the apparatus 300 depicted in FIG. 3 may include additional components and that some of the components described herein may be removed and/or modified without departing from a scope of the apparatus 300.

[0023] As shown in FIG. 3, the apparatus 300 may include the same features as the apparatus 100 depicted in FIG. 1. Particularly, the apparatus 300 may include the processor 120 and the memristor array 130 depicted in FIG. 1. The apparatus 300 is also depicted as including a voltage source 330 and a current reader 340 connected to the processor 120. The voltage source 330 and the current reader 340 may be provided on the substrate 110 or may be located outside of the substrate 110.

[0024] As also shown, the memristor array 130 may have a diagonal cross-bar architecture, which includes a plurality of row lines 310a-310n and a plurality column lines 320a-320n. The variable "n" may represent an integer value greater than one. Each of the row lines 310a-310n and the column lines 320a-320n may be an electrically conductive wire, an electrode, or the like.

[0025] The memristor array 130 is also depicted as having a plurality of memristors 240 electrically connected to respective pairs of row lines 310a-310n and column lines 320a-320n. That is, a first memristor 240 is depicted as being electrically connected to the first row line 310a and the first column line 320a, a second memristor 240 is depicted as being electrically connected to the second row line 310b and the second column line 320b, and an nth memristor 240 is

depicted as being electrically connected to the nth row line 310n and the nth column line 320n. In an example, and although not shown, memristors 240 may be provided at each of the junctions of the row lines 310a-310n and the column lines 320a-320n. In another example, the memristor array 130 includes the memristors 240 as depicted in FIG. 3 and does not include additional memristors between other row line and column line combinations. In this example, for instance, a memristor may not be positioned in electrical connection between the first row line 310a and the second column line 320b, the second row line 310b and the first column line 320a, etc.

[0026] Each of the row lines 310a-310n is depicted as including a respective row input 312a-312n and each of the column lines 320a-320n is depicted as including a respective column input 322a-322n. The voltage source 330, which may be connected to a power supply (not shown), is also depicted as being connected to each of the row inputs 312a-312n and the column inputs 322a-322n. The connections between the voltage source 330 and the row inputs 312a-312n and the column inputs 322a-322n are shown as dotted lines to distinguish those lines from the row lines 310a-310n and the column lines 320a-320n. In any regard, the processor 120 may control or otherwise cause the voltage source 330 to apply voltages at multiple levels to the row inputs 312a-312n and the column inputs 322a-322n. The multiple levels of voltages may include a resetting voltage, a reading voltage, a first voltage, and a second voltage, in which each of these voltages differ from each other.

[0027] The processor 120 may control or otherwise cause the voltage source 330 to supply respective ones of the multiple voltage levels independently to each of the row inputs 312a-312n and the column inputs 322a-322n. That is, the processor 120 may cause the voltage source 330 to apply a first voltage to the first row input 312a and to apply a second voltage to the first column input 322a. In addition, the processor 120 may cause the voltage source 330 to apply voltages to respective pairs of row inputs 312a-322n and column inputs 322a-322n in a sequential manner. That is, the processor 120 may cause the voltage source 330 to apply voltages to the first row input 312a

and the first column input 322a at a first time, apply voltages to the second row input 312b and the second column input 322b at a second time, and so forth. In this regard, the resistance states of the memristors 240 in the memristor array 130 that do not correspond to any characters in the first and second character strings may not be affected during application of the first and second voltages on the memristors 240 that do correspond to the characters in the first and second character strings.

[0028] As discussed in greater detail herein below, the processor 120 may control the voltage source 330 to supply different levels of voltages to the row inputs 312a-312n and the column inputs 322a-322n based upon values assigned to each of the row inputs 312a-312n and the column inputs 322a-322n. As also discussed below, the row inputs 312a-312n may be assigned values that correspond to the characters in a first character string and the column inputs 322a-322n may be assigned values that correspond to the characters in a second character string.

[0029] According to an example, each of the memristors 240 is a unipolar memristor. That is, switching of the memristors 240 may depend on the amplitude but not the polarity of an applied voltage. In addition, the resistance states of the unipolar memristors 240 may be set and reset at the same polarity of the applied voltage. Moreover, states of the memristors 240 may be changed when there is a difference in potential across the memristors 240. That is, if the same voltage is applied to the first row input 312a and the first column input 322a, there will be an equal potential across the memristor 240 that is connected to the first row line 310a and the first column line 320a and thus, no electrical field will be generated across the memristor 240. In this instance, the state of the memristor 240 will not be changed. However, if a first voltage is applied to the first row input 312a and a second voltage is applied to the first column input 322a, there will be a difference in potential across the memristor 240, and thus, the state of the memristor 240 may be changed. The states of the memristors 240 may thus be controlled through the control of the voltages applied to each of the row inputs 312a-312n and the column inputs 322a-322n.

[0030] As discussed above, the different states of the memristors 240 correspond to different resistances in the memristors 240. That is, a memristor 240 in one state may have a different resistance as compared with a memristor 240 in another state. The states of the memristors 240 may thus affect the total current across the memristor array 130. As discussed in greater detail below, following application of the voltages through the memristor array 130 to set the states of the memristors 240 according to the similarities and differences in the characters of the first and second character strings, the processor 120 may cause the voltage source 330 to apply a read voltage across the memristor array 130. As the read voltage is applied, the current reader 340 may determine the total current across the memristors 240 resulting from the applied read voltage. The current reader 340 may also forward the determined total current to the processor 120.

[0031] The processor 120 may calculate the hamming distance between the first character string and the second character string based upon the determined total current. In other words, the processor 120 may calculate the hamming distance between the first character string and the second character string, which are also referred to herein as a first set of symbols and a second set of symbols, based upon the states of the memristors 240. According to an example, the processor 120 may calculate the hamming distance from a predetermined correlation between hamming distances and total current values. That is, a calibration process using test values may be performed to determine the hamming distances, e.g., the number of positions at which the corresponding characters in the first and second character strings differ from each other, corresponding to different total current values and/or the total current values corresponding to different hamming distances .

[0032] Turning now to FIGS. 4 and 5, there are respectively shown flow diagrams of methods 400 and 500 for computing a hamming distance between a first character string and a second character string, according to two examples. It should be understood that the methods 400 and 500 depicted in FIGS. 4 and 5 may include additional operations and that some of the

operations described herein may be removed and/or modified without departing from the scopes of the methods 400 and 500. The descriptions of the methods 400 and 500 are made with reference to the features depicted in FIG. 3 for purposes of illustration and thus, it should be understood that the methods 400 and 500 may be implemented in apparatuses having architectures different from that shown in FIG. 3.

[0033] At block 402, either a first value or a second value may be assigned to each of a plurality of row inputs 312a-312n and to each of a plurality of column inputs 322a-322n of a memristor (e.g., memory device) array 130. That is the processor 120 may assign one of the first value and the second value to each of the row inputs 312a-312n and the column inputs 322a-322n. Particularly, the processor 120 may correlate the characters in a first character string to the positions of the row inputs 312a-312n and may correlate the characters in a second character string to the positions of the column inputs 322a-322n. Thus, for instance, the processor 120 may assign the first character in the first character string to the first row input 312a, the second character in the first character string to the second row input 312b, and so forth. Likewise, the processor 120 may assign the first character in the second character string to the first column input 322a, the second character in the second character string to the second column input 322b, and so forth.

[0034] According to an example, the processor 120 may convert the characters in the first character string and the second character string into Boolean values prior to assigning the characters to the row inputs 312a-312n and the column inputs 322a-322n. That is, the processor 120 may assign the Boolean values of the characters to the row inputs 312a-312n and the column inputs 322a-322n, in which each of the characters is assigned the same number of Boolean values. In this example, each character in the character strings may be converted into multiple Boolean values and each of the Boolean values corresponding to a character may be assigned to multiple ones of the row inputs 312a-312n. Thus, for instance, a first character in the first character string may be converted to three Boolean values "010". In this example, the first Boolean

value "0" may be assigned to the first row input 312a, the second Boolean value "1" may be assigned to the second row input 312b, and the third Boolean value "0" may be assigned to the third row input 312c. The remaining characters in the character strings may be converted in a similar manner. In addition, the processor 120 may assign the Boolean values of the characters in the orders in which the characters are arranged in the first and second character strings.

[0035] Following block 402, each of the characters or the Boolean values corresponding to the characters in the first character string may be assigned to a respective row input 312a-312n according to the order in which the characters are arranged in the first character string. In addition, each of the characters or the Boolean values corresponding to the characters in the second character string may be assigned to a respective column input 322a-322n according to the order in which the characters are arranged in the second character string. Moreover, each of the characters or the Boolean values is one of a first value, e.g., 0, and a second value, e.g., 1.

[0036] At block 404, a first voltage may be caused to be applied to the row inputs 312a-312n and the column inputs 322a-322n that are assigned the first value. In addition, at block 406, a second voltage may be caused to be applied to the row inputs 312a-312n and the column inputs 322a-322n that are assigned the second value. For instance, the processor 120 may cause or control the voltage source 330 to apply the first voltage and the second voltage to each of the row inputs 312a-312n and the column inputs 322a-322n as indicated at blocks 404 and 406 concurrently. In another example, the processor 120 may cause the voltage source 330 to apply voltages to respective pairs of row inputs 312a-322n and column inputs 322a-322n in a sequential manner, as discussed above.

[0037] As also discussed above, the first voltage differs from the second voltage and the second voltage may be relatively larger than the first voltage. By way of example, the first voltage may be a zero voltage and the second voltage may be a voltage that is to generate a sufficiently strong electrical field to change the resistance state of a memristor 240. For instance, the second

voltage may be about 3V or more. In another example, the first voltage may be around 1V and the second voltage may be about 5V or more. It should be understood that these voltages are for illustrative purposes and that the first and second voltages may have other values without departing from a scope of the method 400.

[0038] According to an example, a sufficiently high current compliance may be set on the memristors 240 to enable the resistance states of the memristors 240 to be switched through application of the second voltage.

[0039] Application of the first voltage and the second voltage as indicated at blocks 404 and 406 may either cause the potentials across each of the memristors 240 to respectively be the same or different from each other. The potential across a memristor 240 may be the same if the row input and the column input of the row line and column line to which the memristor 240 is attached are at the same voltage level. The potentials across a memristor 240 may be different from each other if the row input and the column input of the row line and column line to which the memristor 240 is attached are different from each other. As discussed above, the resistance state of the memristor 240 may remain unchanged when a potential difference does not exist across the memristor 240. However, when a potential difference does exist across the memristor 240, the resistance state of the memristor 240 may be changed, e.g., may be increased. That is, the resistance state of a memristor 240 may be changed if the voltage applied to the row input of the memristor 240 is the first voltage, e.g., zero volts, and the voltage applied to the column input of the memristor 240 is the second voltage.

[0040] Thus, the resistance states of the memristors 240 may remain the same for those memristors 240 positioned between row and column inputs respectively corresponding to characters in the first and second character strings that are the same. Likewise, the resistance states of the memristors 240 may be changed for those memristors 240 positioned between row and column inputs respectively corresponding to characters in the first and second character strings that differ from each other.

[0041] According to an example, a resetting voltage may be applied to the memristors 240 prior to implementation of blocks 404 and 406 to reset the memristors 240 such that all of the memristors 240 are in a common resistance state. For instance, the processor 120 may cause the voltage source 330 to apply a resetting voltage to each of the row inputs 312a-312n without applying a voltage on each of the column inputs 322a-322n, or vice versa. According to an example, the processor 120 may cause the voltage source 330 to apply a relatively larger voltage as compared with the second voltage to each of the row inputs 312a-312n to thus cause each of the memristors 240 to have the same states. In this example, the processor 120 may assign the same values to each of the row inputs 312a-312n (or column inputs 322a-322n), in which the same values are higher than the second voltage.

[0042] At block 408, a read voltage may be caused to be applied across the memristors 240 (or equivalently, memory devices 240). For instance, the processor 120 may cause the voltage source 330 to apply the read voltage across the memristor 240 through the row inputs 312a-312n and the column inputs 322a-322n. The read voltage may be lower than the second voltage to prevent the read voltage from changing the states of the memristors 240 but may be of sufficient level to enable a total current through the memristors 240 to be determined.

[0043] At block 410, a total current across the memristors 240 resulting from the applied read voltage may be determined. For instance, the processor 120 may determine the total current from the current reader 340, which may be an ammeter. The total current across the memristors 240 may vary depending upon the number of memristors 240 whose resistance states have been modified during implementation of blocks 404 and 406.

[0044] At block 412, a hamming distance between the first character string and the second character string may be calculated. For instance, the processor 120 may calculate the hamming distance between the first character string and the second character string based upon the determined total current across the memristors 240. Particularly, the processor 120 may compare the

determined total current with predetermined correlations between total current values and hamming distances and identify the hamming distance that corresponds to the determined total current.

[0045] With reference now to FIG. 5, at block 502, a first character string and a second character string, which are also referenced herein as a first set of symbols and a second set of symbols, may be accessed. For instance, the processor 120 may access the first character string and the second character string, in which the hamming distance between the first character string and the second character string is to be determined. The characters in the first character string and the second character string may be symbols, numerals, letters, combinations of different character types, or the like.

[0046] At block 504, the characters in the first character string and the second character string may optionally be converted to Boolean values, or equivalently, to another binary code. Block 504 may be optional, for instance, not performed in cases in which the characters in the first character string and the second character string are already in binary code form. The processor 120 may convert the characters into binary code form in any of the manners discussed above with respect to the method 400 in FIG. 4.

[0047] At block 506, a resetting voltage may be caused to be applied to the memristors 240. For instance, the processor 120 may cause the voltage source 330 to apply the resetting voltage to either or both of the row inputs 312a-312n and the column inputs 322a-322n to reset the memristors 240 such that all of the memristors 240 are in the same resistance state.

[0048] Following conversion of the characters, as indicated at block 508, the method 400 may be implemented to determine the hamming distance between the first character string and the second character string.

[0049] Some or all of the operations set forth in the methods 400 and 500 may be contained as utilities, programs, or subprograms, in any desired computer accessible medium. In addition, the methods 400 and 500 may be embodied by computer programs, which may exist in a variety of forms both

active and inactive. For example, they may exist as machine readable instructions, including source code, object code, executable code or other formats. Any of the above may be embodied on a non-transitory computer-readable storage medium.

[0050] Examples of non-transitory computer-readable storage media include computer system RAM, ROM, EPROM, EEPROM, and magnetic or optical disks or tapes. It is therefore to be understood that any electronic device capable of executing the above-described functions may perform those functions enumerated above.

[0051] Turning now to FIG. 6, there is shown a schematic representation of a computing device 600, which may implement the apparatus 100, 300 depicted in FIGS. 1 and 3, according to an example. The computing device 600 may include an apparatus 602 and an input/output interface 604. The apparatus 602 may include the processor 120 and the memristor array 130 as shown in FIGS. 1 and 3. The input/output interface 604 may provide an interface with an input device, such as a keyboard, a mouse, etc., and an output device, such as a display. The computing device 600 may also include a network interface 608, such as a Local Area Network LAN, a wireless 802.11x LAN, a 3G mobile WAN or a WiMax WAN, through which the computing device 600 may connect to a network (not shown). The computing device 600 may further include a computer-readable medium 610 on which is stored sets of machine-readable instructions. Each of these components may be operatively coupled to a bus 612, which may be an EISA, a PCI, a USB, a FireWire, a NuBus, a PDS, or the like.

[0052] The computer-readable medium 610 may be any suitable medium that participates in providing instructions to the processor 602 for execution. For example, the computer-readable medium 610 may be non-volatile media, such as an optical or a magnetic disk; volatile media, such as memory. In an example, the computer-readable medium 610 is another memristor array and is fabricated on the apparatus 602 such that the processor 120 may access the computer-readable medium 610 directly, i.e., without going through the bus 612.

As shown, the computer-readable medium 610 may store a hamming distance computing module 620, which the processor 120 may implement to compute the hamming distance between character strings. The hamming distance computing module 620 may thus be a set of machine readable instructions pertaining to one of the methods 400 and 500.

[0053] Although described specifically throughout the entirety of the instant disclosure, representative examples of the present disclosure have utility over a wide range of applications, and the above discussion is not intended and should not be construed to be limiting, but is offered as an illustrative discussion of aspects of the disclosure.

[0054] What has been described and illustrated herein are examples of the disclosure along with some variations. The terms, descriptions and figures used herein are set forth by way of illustration only and are not meant as limitations. Many variations are possible within the scope of the disclosure, which is intended to be defined by the following claims -- and their equivalents -- in which all terms are meant in their broadest reasonable sense unless otherwise indicated.

CLAIMS

What is claimed is:

1. An apparatus for computing hamming distance, said apparatus comprising:

a memristor array formed of a plurality of memristors, wherein each of the memristors is individually addressable through a row input and a column input;

a processor to:

assign one of a first value and a second value to each of the row inputs and the column inputs of the plurality of memristors, wherein the row inputs correspond to characters in a first character string and the column inputs correspond to characters in a second character string;

cause a first voltage to be applied to the row inputs and the column inputs that are assigned the first value and a second voltage to be applied to the row inputs and the column inputs that are assigned the second value, wherein states of the memristors are to be changed when voltages applied to the row inputs and the column inputs of the memristors differ from each other and wherein states of the memristors are to remain unchanged when the first and second voltages are the same; and

calculate a hamming distance between the first character string and the second character string based upon the states of the memristors.

2. The apparatus according to claim 1, wherein the processor is further to cause a read voltage to be applied across the plurality of memristors and to determine a total current across the plurality of memristors resulting from the applied read voltage.

3. The apparatus according to claim 2, wherein the processor is further to calculate the hamming distance from a predetermined correlation between hamming distances and total current values.

4. The apparatus according to claim 1, wherein the processor is further to:
convert the characters in the first character string and the second character string into Boolean values, wherein the first value and the second value are the Boolean values.
5. The apparatus according to claim 1, further comprising:
a substrate, wherein the memristor array and the processor are attached to the substrate.
6. The apparatus according to claim 5, further comprising:
another memristor array attached to the substrate, wherein the memristor array is to function solely as an accelerator for the processor to accelerate calculation of hamming distances and wherein the processor is to store data in the another memristor array.
7. The apparatus according to claim 1, wherein the processor is further to cause a resetting voltage to be applied to one or both of the row inputs and the column inputs prior to application of the first voltage and the second voltage, wherein application of the resetting voltage is to cause the memristors to be in a common resistance state with respect to each other.
8. The apparatus according to claim 1, wherein the memristors are unipolar memristors arranged in a diagonal cross-bar architecture.
9. The apparatus according to claim 1, wherein a sufficiently high current compliance is set on the memristors to enable the states of the memristors to be switched through application of the second voltage.
10. A method for computing a hamming distance between a first set of symbols and a second set of symbols, said method comprising:
assigning either a first value or a second value to each of a plurality of row inputs and to each of a plurality of column inputs of a memory device array,

wherein the plurality of row inputs correspond to the symbols in the first set of symbols and the plurality of column inputs corresponds to the symbols in the second set of symbols;

causing a first voltage to be applied to the row inputs and the column inputs that are assigned the first value and a second voltage to be applied to the row inputs and the column inputs that are assigned the second value, wherein a resistance state of a memory device is to be changed when the voltages applied to the row input and the column input of the memory device differ from each other and wherein the resistance state of the memory device is to remain unchanged when the voltages are the same;

causing a read voltage to be applied across the memory devices;

determine a total current across the memory devices resulting from the read voltage; and

calculating a hamming distance between the first set of symbols and the second set of symbols based upon the determined total current.

11. The method according to claim 10, wherein calculating the hamming distance further comprises calculating the hamming distance from a predetermined correlation between hamming distances and total current values.

12. The method according to claim 10, further comprising:

converting the symbols in the first set of symbols and the symbols in the second set of symbols into Boolean values, wherein the first value and the second value are the Boolean values.

13. The method according to claim 10, wherein causing the first voltage to be applied to the row inputs and the column inputs that are assigned the first value further comprises causing a zero voltage to be applied as the first voltage.

14. A system for determining a difference between a first character string and a second character string, said system comprising:

a memristor array formed of row lines, column lines, and memristors connected to respective pairs of row lines and column lines;

a processor to:

assign each of the characters in the first character string and the second character string to one of a first value and a second value;

assign each of the characters in the first character string to a respective row line and each of the characters in the second character string to a respective column line;

assign one of the first value and the second value to each of the row lines and the column lines according to the assignment of the characters to the row lines and the column lines;

designate a first voltage to be applied to the row inputs and the column inputs that are assigned the first value and a second voltage to be applied to the row inputs and the column inputs that are assigned the second value;

determine a total current across the memristor array; and

calculate a hamming distance between the first character string and the second character string corresponding to the determined total current.

15. The system according to claim 14, further comprising a substrate, wherein the memristor array the processor are attached to the substrate.

1 / 5

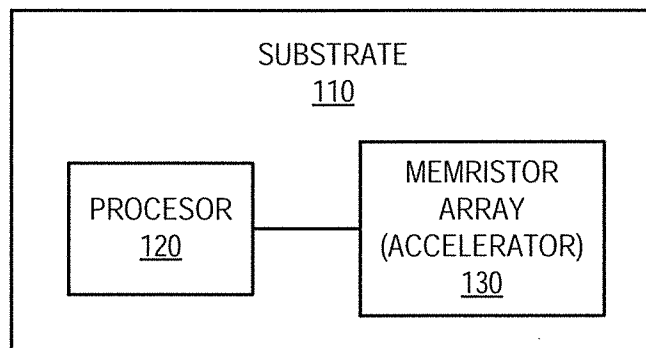
APPARATUS
100 →

FIG. 1

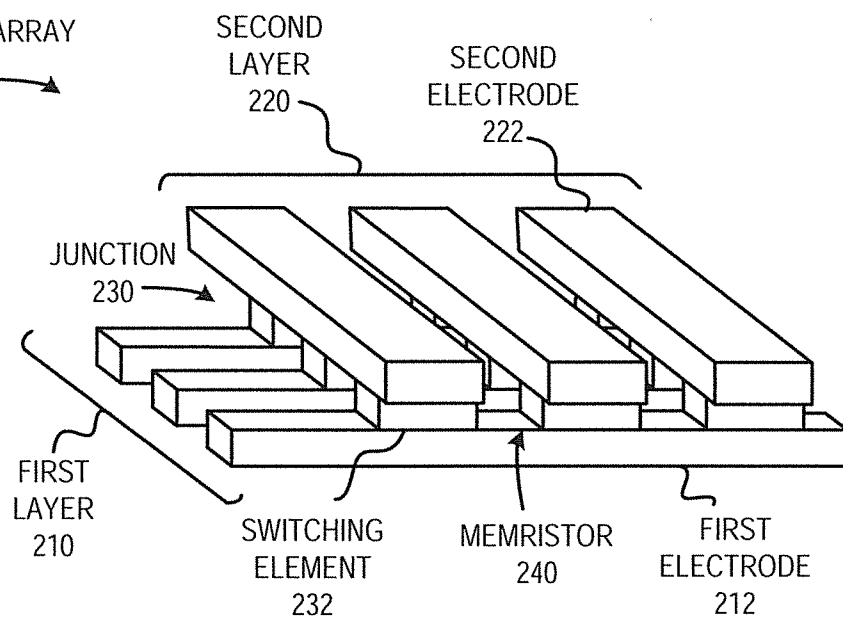
MEMRISTOR ARRAY
130 →

FIG. 2

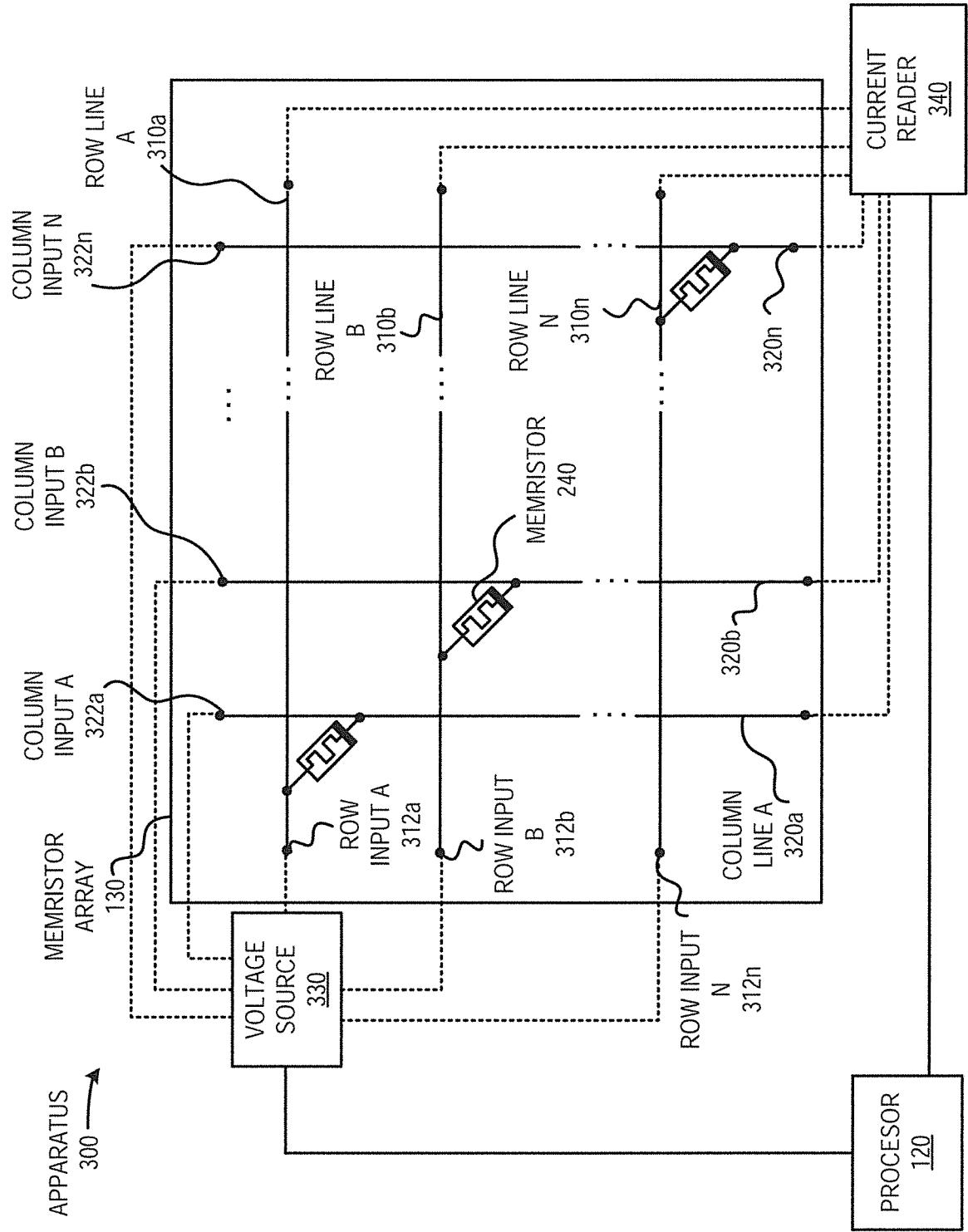
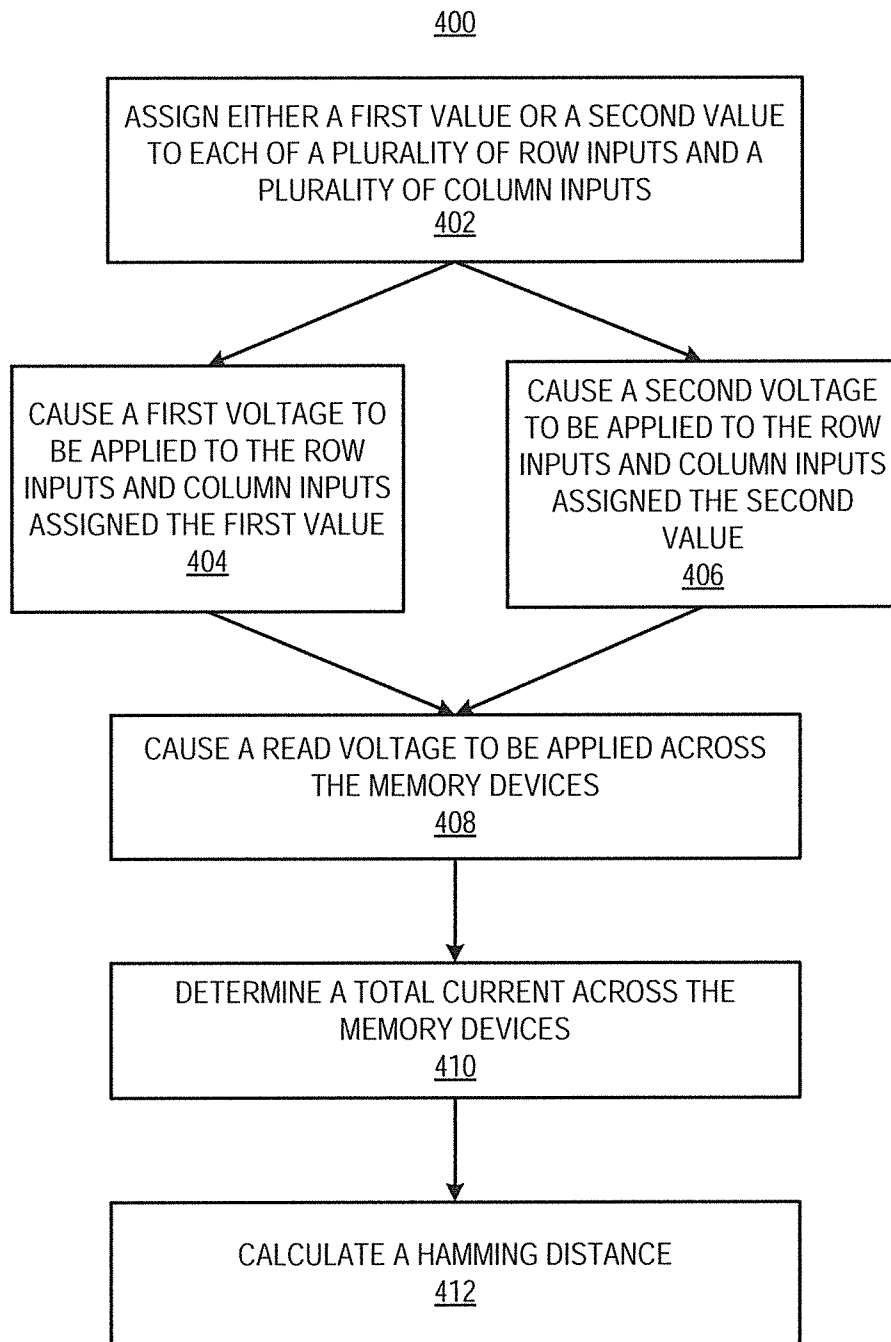
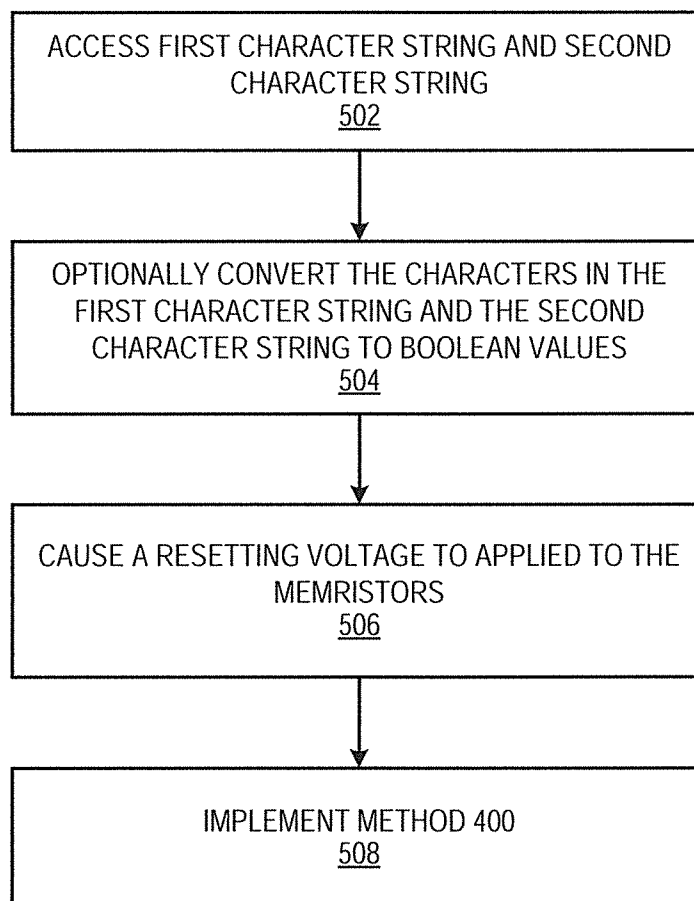


FIG. 3

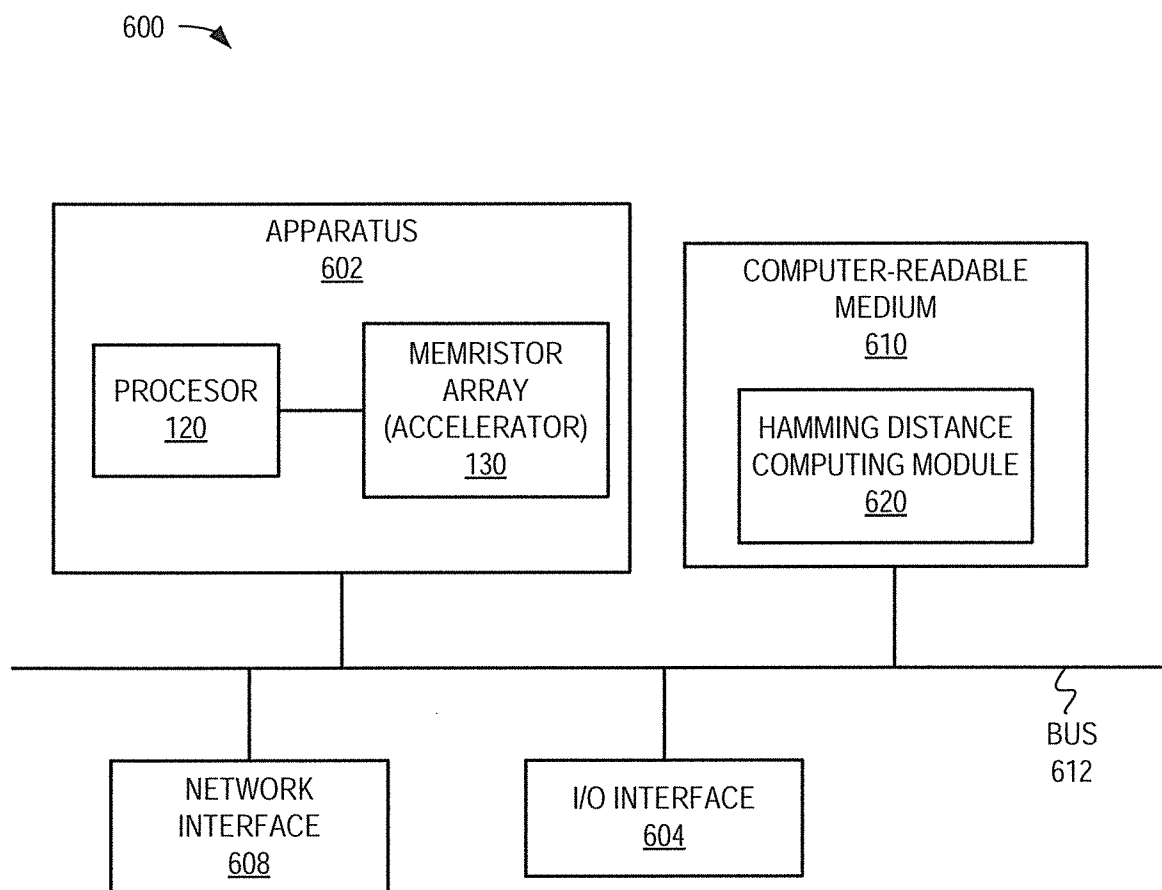
3 / 5

**FIG. 4**

4 / 5

500**FIG. 5**

5 / 5

**FIG. 6**

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US2015/013508**A. CLASSIFICATION OF SUBJECT MATTER****G11C 13/00(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 13/00; G11C 13/02; G01R 31/28; H03M 13/05; H03M 13/19; G11C 11/00

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: hamming distance, memristor, voltage, character string, value, state of memristor, crossbar array memory, read voltage

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	EP 2074624 B1 (HEWLETT-PACKARD DEVELOPMENT COMPANY, L.P.) 03 March 2010 See paragraphs [0017], [0023], [0027]-[0029], [0033]-[0039], [0078]; and figures 1, 2A-2B.	1-15
A	US 2013-0019143 A1 (SAMSUNG ELECTRONICS CO., LTD.) 17 January 2013 See paragraphs [0030]-[0031]; and figure 1.	1-15
A	US 2014-0189472 A1 (ALEXANDER GENDLER et al.) 03 July 2014 See paragraphs [0014], [0017]; and figure 1.	1-15
A	US 2014-0092667 A1 (HEWLETT-PACKARD DEVELOPMENT COMPANY) 03 April 2014 See paragraph [0064]; and figure 4A.	1-15
A	US 2003-0182603 A1 (ROBERT B. GASS) 25 September 2003 See paragraphs [0061]-[0062]; and figure 6.	1-15



Further documents are listed in the continuation of Box C.



See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search

14 October 2015 (14.10.2015)

Date of mailing of the international search report

19 October 2015 (19.10.2015)

Name and mailing address of the ISA/KR

International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon Metropolitan City, 35208,
Republic of Korea

Facsimile No. +82-42-472-7140

Authorized officer

BYUN, Sung Cheal

Telephone No. +82-42-481-8262



INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No.

PCT/US2015/013508

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
EP 2074624 B1	03/03/2010	AT 459963 T	15/03/2010
		CN 101529523 A	09/09/2009
		CN 101529523 B	25/07/2012
		EP 2074624 A2	01/07/2009
		JP 05086359 B2	28/11/2012
		JP 2010-507184 A	04/03/2010
		KR 10-1395170 B1	16/05/2014
		KR 10-2009-0068373 A	26/06/2009
		TW 200828508 A	01/07/2008
		US 2008-0089110 A1	17/04/2008
		US 7778061 B2	17/08/2010
		WO 2008-048597 A2	24/04/2008
		WO 2008-048597 A3	19/06/2008
US 2013-0019143 A1	17/01/2013	KR 10-1403314 B1	05/06/2014
		US 2009-0292973 A1	26/11/2009
		US 8301978 B2	30/10/2012
		US 8543892 B2	24/09/2013
US 2014-0189472 A1	03/07/2014	US 8826107 B2	02/09/2014
US 2014-0092667 A1	03/04/2014	US 8861256 B2	14/10/2014
US 2003-0182603 A1	25/09/2003	US 6909358 B2	21/06/2005