

(19) World Intellectual Property Organization
International Bureau



(43) International Publication Date
3 June 2010 (03.06.2010)

(10) International Publication Number
WO 2010/062302 A1

- (51) **International Patent Classification:**
G1C 8/00 (2006.01) *G1C 7/00* (2006.01)
- (21) **International Application Number:**
PCT/US2009/005600
- (22) **International Filing Date:**
14 October 2009 (14.10.2009)
- (25) **Filing Language:** English
- (26) **Publication Language:** English
- (30) **Priority Data:**
12/259,380 28 October 2008 (28.10.2008) US
- (71) **Applicant (for all designated States except US):** MICRON TECHNOLOGY, INC. [US/US]; 8000 S. Federal Way, Boise, ID 83716-9632 (US).
- (72) **Inventor; and**
- (75) **Inventor/Applicant (for US only):** MANNING, Troy [US/US]; 4753 S. Ten Mile Road, Meridian, ID 83642 (US).
- (74) **Agents:** KERN, Jacob, T. et al; Brooks, Cameron & Huebsch, PLLC, 1221 Nicollet Avenue, Suite 500, Minneapolis, MN 55403 (US).
- (81) **Designated States (unless otherwise indicated, for every kind of national protection available):** AE, AG, AL, AM,

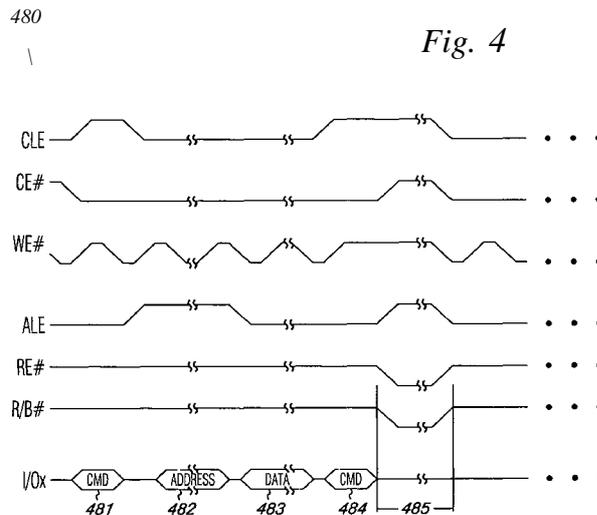
AO, AT, AU, AZ, BA, BB, BG, BH, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PE, PG, PH, PL, PT, RO, RS, RU, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, 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, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European (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, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, 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:** LOGICAL UNIT OPERATION



(57) **Abstract:** The present disclosure includes methods and devices for logical unit operation. One device embodiment includes a number of logical units, wherein each of the number of logical units has a unique address. The device includes control circuitry coupled to the number of logical units and configured optionally to control more than one of the number of logical units with one of a number of commands and one address.

WO 2010/062302 A1

LOGICAL UNIT OPERATION

Technical Field

[0001] The present disclosure relates generally to semiconductor memory devices, methods, and systems, and more particularly, to logical unit operation.

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), phase change random access memory (PCRAM), and flash memory, among others.

[0003] Solid state memory devices are utilized as volatile and non-volatile memory for a wide range of electronic applications. Flash memory, which is just one type of solid state memory, typically uses a one-transistor memory cell that allows for high memory densities, high reliability, and low power consumption.

[0004] Solid state memory devices, including flash devices, can be combined together to form a solid state drive. A solid state drive can be used to replace hard disk drives as the main storage device for a computer, as the solid state drives can have advantages over hard drives in terms of performance, size, weight, ruggedness, operating temperature range, and power consumption. For example, flash solid state drives can have superior performance when compared to magnetic disk drives due to their lack of moving parts, which may ameliorate seek time, latency, and other electro-mechanical delays associated with magnetic disk drives.

[0005] A solid state drive is a data storage device that uses solid state memory to store persistent data. A solid state drive may include NAND flash non-volatile memory and/or DRAM volatile memory. Solid state drive manufacturers can use nonvolatile flash memory to create a drive that does not use an internal battery supply thus allowing the drive to be more versatile and

compact. Solid state drives using flash memory, also known as flash drives, can use standard disk drive form factors (e.g., 1.8-inch, 2.5-inch, and 3.5-inch, among others).

[0006] For some storage applications, hard drives may be arranged in a redundant array, such as a redundant array of inexpensive disks, also referred to as a redundant array of independent disks (RAID). A RAID can refer to data storage systems that can divide and replicate data among multiple hard disk drives. Such arrangements, e.g., in servers, may be intended to prevent data loss in case a particular drive fails. However, this practice may involve a substantial investment in extra capacity that is not realized in system storage capacity. For example, in some storage applications, as little as 5-30% of a hard disk capacity is used for storage in order to increase performance by reducing seek times associated with the hard disk(s). Furthermore, hard drive failure rates may tend to increase as the drive ages. Solid state devices, conversely, may tend to fail early in life, if at all, and then operate correctly through the end of their expected service life.

Brief Description of the Drawings

[0007] Figure 1 is a functional block diagram of an electronic memory system having at least one memory device operated in accordance with one or more embodiments of the present disclosure.

[0008] Figure 2A illustrates a block diagram of logical unit control in accordance with one or more embodiments of the present disclosure.

[0009] Figure 2B illustrates a block diagram of a memory architecture in accordance with one or more embodiments of the present disclosure.

[0010] Figure 3 illustrates a prior art timing diagram of various signals during operation of a memory device.

[0011] Figure 4 illustrates a timing diagram of various signals during operation of a memory device in accordance with one or more embodiments of the present disclosure.

[0012] Figure 5 illustrates a table including an address data arrangement organized into 5 cycles in accordance with one or more embodiments of the present disclosure.

[0013] Figure 6 illustrates a block diagram of a solid state drive in accordance with one or more embodiments of the present disclosure.

[0014] Figure 7 is a functional block diagram of a memory module having at least one memory device in accordance with one or more embodiments of the present disclosure.

Detailed Description

[0015] The present disclosure includes methods and devices for logical unit operation. One device embodiment includes a number of logical units, wherein each of the number of logical units has a unique address. The device includes control circuitry coupled to the number of logical units and configured optionally to control more than one of the number of logical units with one of a number of commands and one address.

[0016] In the following detailed description of the present disclosure, reference is made to the accompanying drawings that form a part hereof, and in which is shown by way of illustration how one or more embodiments of the disclosure may be practiced. These embodiments are described in sufficient detail to enable those of ordinary skill in the art to practice the embodiments of this disclosure, and it is to be understood that other embodiments may be utilized and that process, electrical, and/or structural changes may be made without departing from the scope of the present disclosure. As used herein, the designators "N," "P," "L," and "B," particularly with respect to reference numerals in the drawings, indicates that a number of the particular feature so designated can be included with one or more embodiments of the present disclosure.

[0017] The figures herein follow a numbering convention in which the first digit or digits correspond to the drawing figure number and the remaining digits identify an element or component in the drawing. Similar elements or components between different figures may be identified by the use of similar digits. For example, 110 may reference element "10" in Fig. 1, and a similar element may be referenced as 210 in Fig. 2. As will be appreciated, elements shown in the various embodiments herein can be added, exchanged, and/or eliminated so as to provide a number of additional embodiments of the present disclosure. In addition, as will be appreciated, the proportion and the relative

scale of the elements provided in the figures are intended to illustrate the embodiments of the present invention, and should not be taken in a limiting sense.

[0018] Figure 1 is a functional block diagram of an electronic memory system 100 having at least one memory device 120 operated in accordance with one or more embodiments of the present disclosure. Memory system 100 includes a processor 110 coupled to a non-volatile memory device 120 that includes a memory array 130 of non-volatile cells. The memory system 100 can include separate integrated circuits or both the processor 110 and the memory device 120 can be on the same integrated circuit. The processor 110 can be a microprocessor or some other type of controlling circuitry such as an application-specific integrated circuit (ASIC).

[0019] The memory device 120 includes an array of non-volatile memory cells 130, which can be floating gate flash memory cells with a NAND architecture. The control gates of memory cells of a "row" are coupled with a word line, while the drain regions of the memory cells of a "column" are coupled to bit lines. The source regions of the memory cells are coupled to source lines. As will be appreciated by those of ordinary skill in the art, the manner of connection of the memory cells to the bit lines and source lines depends on whether the array is a NAND architecture, a NOR architecture, an AND architecture, or some other memory array architecture.

[0020] The embodiment of Figure 1 includes address circuitry 140 to latch address signals provided over I/O connections 162 through I/O circuitry 160. Address signals are received and decoded by a row decoder 144 and a column decoder 146 to access the memory array 130. In light of the present disclosure, it will be appreciated by those skilled in the art that the number of address input connections depends on the density and architecture of the memory array 130 and that the number of addresses increases with both increased numbers of memory cells and increased numbers of memory blocks and arrays.

[0021] The memory device 120 senses data in the memory array 130 by sensing voltage and/or current changes in the memory array columns using sense/buffer circuitry that in this embodiment can be read/latch circuitry 150. The read/latch circuitry 150 can read and latch a page, e.g., a row, of data from the memory array 130. I/O circuitry 160 is included for bi-directional data

communication over the I/O connections 162 with the processor 110. Write circuitry 155 is included to write data to the memory array 130.

[0022] Control circuitry 170 decodes signals provided by control connections 172 from the processor 110. These signals can include chip signals, write enable signals, and address latch signals that are used to control the operations on the memory array 130, including data sensing, data write, and data erase operations, as described herein. In one or more embodiments, the control circuitry 170 is responsible for executing instructions from the processor 110 to perform the operations according to embodiments of the present disclosure. The control circuitry 170 can be a state machine, a sequencer, or some other type of controller. It will be appreciated by those skilled in the art that additional circuitry and control signals can be provided, and that the memory device detail of Figure 1 has been reduced to facilitate ease of illustration.

[0023] Figure 2A illustrates a block diagram of logical unit control in accordance with one or more embodiments of the present disclosure. As used herein, a logical unit can include a group of memory cells, e.g., a die of memory cells, that share a number of control inputs. Control inputs are described in connection with Figures 3 and 4, but can generally include address latch enable (ALE), chip enable (CE), read enable (RE), ready/busy (R/B), write protect (WP), and input/output (I/O) connections such as pins, pads, or the like.

[0024] The embodiment illustrated in Figure 2A includes control circuitry 202, e.g., control circuitry 102 in Figure 1, coupled to a number of logical units 270-A, 270-B, 270-C, 270-D, 270-E, and 270-F. With respect to the embodiment illustrated in Figure 1, the number of logical units 270-A, 270-B, 270-C, 270-D, 270-E, and 270-F can be located in one or more memory arrays 130. Control circuitry 202 is coupled to logical units 270-A and 270-B with control inputs 241-1. Control circuitry 202 is coupled to logical units 270-C, 270-D, 270-E, and 270-F with control inputs 241-2. Logical units 270-A and 270-B can be individual dice, while logical units 270-C and 270-D, as well as logical units 270-E and 270-F can be stacked dice. Each die can include one or more arrays of memory cells.

[0025] In one or more embodiments of the present disclosure, control circuitry 202 can be configured to control more than one logical unit with one command, e.g., a single command. For example, control circuitry 202 can

control logical units 270-A and 270-B with one command issued across control inputs 241-1. As another example, control circuitry 202 can control logical units 270-C, 270-D, 270-E, and 270-F with one command issued across control inputs 241-2. In one or more embodiments, control circuitry 202 can issue a command across control inputs 241-2 such that it controls any one, two, three, or all four logical units 270-C, 270-D, 270-E, and 270-F. Details of such commands and controls are described in more detail herein.

[0026] Figure 2B illustrates a block diagram of a memory architecture in accordance with one or more embodiments of the present disclosure. The memory architecture shown in Figure 2B includes a number of logical units such as "Logical Unit 0" 270-0, "Logical Unit 1" 270-1, . . . , "Logical Unit L" 270-L. Logical units 270-0, 270-1, . . . , 270-L can be analogous to the logical units 270-A, 270-B, 270-C, 270-D, 270-E, and 270-F illustrated in Figure 2A. Each logical unit can include a number of blocks, such as "Block 0" 271 illustrated in Logical Unit 0 270-0. Each block can include a number of pages of memory cells, such as "Page 0" 273 illustrated in Block 0 of Logical Unit 0 270-0. Although not illustrated in Figure 2B, a memory device can also include a number of planes, dice, and other groups of memory cells. As an example, a 128 GB memory device can include 43 14 bytes of data per page, 128 pages per block, 2048 blocks per plane, and 16 planes per device.

[0027] Each page of memory cells can include a number of memory cells, e.g., non-volatile memory cells. As used herein, a page of memory cells means a number of memory cells which can store an amount of data programmable at one time. As an example, the amount of data programmable at one time can be referred to as a page of data, and the memory cells which store the page of data can be referred to as a page of memory cells. In one or more embodiments, a page of memory cells can include memory cells coupled to a particular access line, e.g., word line. In one or more embodiments, memory cells coupled to a particular word line can be divided into more than one page, e.g., into an "even" page and an "odd" page of data. In one or more embodiments, a page of memory cells can include memory cells coupled to more than one word line. As used herein, a block of memory cells means a number of memory cells which can store an amount of data erasable at one time. For example, the amount of data erasable at one time can be referred to as a block of

data, and the memory cells which store the block of data can be referred to as a block of memory cells.

[0028] Figure 3 illustrates a prior art timing diagram 380 of various signals during operation of a memory device. More particularly, timing diagram 380 illustrates signals associated with a programming operation, e.g., a write operation, for writing data redundantly to two logical units that share control signals CLE, CE#, WE#, ALE, RE#, and R/B#. Signals can include a command latch enable (CLE) signal, a chip enable (CE#) signal, a write enable (WE#) signal, an address latch enable (ALE) signal, a read enable (RE#) signal, and a ready/busy (R/B#) signal. Information communicated across input/output circuitry, e.g., an input/output bus, is illustrated by I/Ox. As one of ordinary skill in the art will appreciate, the "#" symbol indicates a particular signal being active in a LOW logic state.

[0029] The CLE signal can be used to load a command from a bus into a command register. The CE# signal can enable or disable a logical unit, e.g., a die, of memory. The WE# signal can transfer commands, addresses, and serial data from a memory access device (e.g., processor, memory controller, control circuitry, host system, etc.), to the memory. For example, the WE# signal can be used to provide a timing reference to a memory device. Timing chains that are activated by the control signals (e.g., ALE and CLE) can be used to control the timing of communications transfers. The memory access device can use control signals to indicate to the memory device when requests for data transactions are sent. The ALE signal can be used to load an address from a bus into an address register. The RE# signal can transfer serial data from the memory to a host system.

[0030] Various control signals coordinate communication of a command, address and other information, and data across a memory interface. According to the previous approach illustrated in Figure 3, a programming operation can include a data input command 381-1, e.g., serial data input, sent across the I/O bus, followed by address information 382-1, e.g., a number of address cycles 382-1 containing information to identify a particular logical unit or other location in memory, and then by the data 383-1. In the illustration of Figure 3, address information 382-1 identifies a first logical unit. Address information communicated during the address cycle 382-1, can be latched in an address

register and further directed to a column decode and/or a row decode, which in turn, drives selection of one or more memory cells. Data 383-1 can be written to or read from memory through a cache register and data register. Control logic can load status information into a status register, which may be further communicated to I/O control.

[0031] According to the previous approach illustrated in Figure 3, once the data has been input, e.g., to a register, a program command 384-1 can issue to begin programming of the data to the location(s) included in the address 382-1, e.g., from the register to a location in memory. Timing diagram 380 includes an indication of the time 385-1 during which data is programmed to the memory device. Such programming time 385-1 can be associated with a time during which the ready/busy signal R/B# associated with the logical unit to which data is being programmed being LOW. Although not illustrated in timing diagram 380, a programming operation may be followed by one or more program verify operations, read status operations, or other operations to facilitate improved data reliability.

[0032] As described above, timing diagram 380 includes signals associated with programming data redundantly to two logical units. Accordingly, after data is programmed to the first logical unit during time 385-1, a second serial data input command 381-2 can be issued, accompanied by address information 382-2, which identifies the second logical unit, and the redundant copy of data 383-2. Data 383-2 can be followed by programming command 384-2 to program the redundant copy of data to the second logical unit during programming time 385-2. As illustrated in timing diagram 380, some prior art approaches to programming data redundantly to two logical units involve issuing two data input commands and two sets of address information (for the two logical units), clocking data across an I/O bus twice, issuing two programming commands, and programming the data to two different logical units at separate times.

[0033] Figure 4 illustrates a timing diagram 480 of various signals during operation of a memory device in accordance with one or more embodiments of the present disclosure. More particularly, Figure 4 illustrates signals associated with a programming, e.g., write, operation for writing data redundantly to two

logical units that share control signals CLE, CE#, WE#, ALE, RE#, and R/B#.

Such signals can be analogous to those described with respect to Figure 3.

[0034] According to one or more embodiments of the present disclosure, more than one logical unit can be controlled with one of a number of commands, e.g., a single programming command. Thus, in the embodiment illustrated in Figure 4, a data input command 481 can be sent across the I/O bus followed by address information 482 and data 483. The data 483 can be followed by one programming command 484, which can program the data 483 during programming time 485 to both first and second logical units, and is indicated by R/B# being LOW during programming time 485. Programming time 485 for programming data to two or more logical units can be substantially similar or equal to programming time 385-1 illustrated in Figure 3 for programming data to one logical unit, assuming that the same data, or same amount of data, is being programmed.

[0035] Address information 482 can identify a location in memory, e.g., a logical unit. An example of information included with address information 482 is illustrated in Figure 5. Address information can be transferred across the I/O bus to an address register in a number of cycles, e.g., 8 bit or 16 bit cycles, however embodiments are not limited to a particular type or amount of data transfer of the address information 482. For embodiments that cycle address information, each cycle can be associated with a rising edge of the WE# signal. Timing diagram 480 includes a break in the illustration of address information 482, and corresponding portions of other signals, to illustrate that a number of address cycles can be used to indicate a location in memory. Although the address information 482 corresponds to only one logical unit, more than one logical unit can be controlled by command 484, e.g., a single programming command, which is associated with the address.

[0036] In one or more embodiments, more than one logical unit can be controlled by one or more commands associated with the address information 482 when control circuitry is configured to mask at least a portion of the address 482. In one or more embodiments, more than one logical unit can be controlled without regard to at least the portion of the address 482. The portion of the address can be a portion of the address that identifies the logical unit. As described below in connection with Figure 5, address information 482 can

include a number of portions that identify locations in memory, e.g., a logical unit, a die, a plane, a block, a page, or some other portion of memory. In one or more embodiments, the number of logical units that are controlled by the one or more commands associated with the address information 482 can be controlled simultaneously.

[0037] In one or more embodiments, the ability of a memory device to control more than one logical unit with one or more commands can be optionally enabled. An operating parameter can be set, e.g., programmed, in the memory device, e.g., during initialization of the device. For example, a "set features" command can be used to alter default power-on behavior of the memory device. The set features command can be used to program an operating parameter into a location in memory that is reserved for such information. In one or more embodiments, such operating parameters can be stored in the device until it is powered down. Setting the operating parameter can be similar to programming data, e.g., as illustrated by timing diagram 480. That is a set features command can issue, followed by address information, which can identify a portion of the memory device for which a particular operating parameter is being set. The address information can be followed by data, e.g., values representing a particular setting for the operating parameter. Such information can be clocked in through the I/O and programmed into an appropriate memory location. Once programmed, the operating parameter can be read by the device to determine operating behavior.

[0038] Embodiments are not limited to programming an operating parameter during device initialization, and such an operating parameter can be changed without powering down the device, e.g., a different operating parameter can be programmed to change the behavior of the device. For example, it may be beneficial to enable a memory device to operate more than one logical unit as controlled by a single command so that data can be programmed redundantly until a threshold of used space is met, then disable such redundant programming thereafter. Such uses of this feature are described in more detail in commonly assigned US Patent Application number 12/259,363 attorney docket number 1002.0350001, entitled "Solid State Drive Operation," including at least one common inventor, Troy Manning, filed on the same date herewith. Such embodiments can include setting the operating parameter to a first value and

operating more than one logical unit for certain commands. Subsequently, the operating parameter can be set to a second value and the device can be operated such that there only one logical unit is controlled by each command.

[0039] In one or more embodiments, when the particular operating parameter is set such that one or more commands can control more than one logical unit, certain commands may continue to control only one logical unit. For example, programming commands, erasing commands, data move commands, cached commands, and multiple plane commands can control more than one logical unit. However, reading commands, status commands, get feature commands, and set feature commands may control only one logical unit, per their associated address information associated with such commands. Such embodiments can be useful in allowing data to be programmed to, moved from, or erased from more than one logical unit for redundancy, among other purposes. Such embodiments can also be useful in allowing one copy of the data to be read during a sensing operation to reduce the operating overhead that would be associated with sensing more than one copy of the same data.

[0040] Figure 5 illustrates a table 590 including an address data arrangement organized into 5 cycles in accordance with one or more embodiments of the present disclosure. The data arrangement shown in Figure 5 includes information transmitted during the address cycles (e.g., address information 482 in Figure 4) arranged into five 8-bit portions (e.g., five address cycles "First, second, . . . , fifth"). The data arrangement shown in Figure 5 may be communicated across an 8-bit bus using five address cycles. Embodiments of the present disclosure are not limited to this configuration, and may be arranged to have a different word length (e.g., 16 bits), including more or less bits per word so as to comprise more or fewer address cycles.

[0041] The row labeled "CYCLE" includes headers for each of the 8 I/O bits for each address cycle. The designator "CA" indicates a column address, "PA" indicates a plane address, and "BA" indicates a block address. The data arrangement shown in table 590 includes a number of unused bits (shown set LOW in Figure 5). In the example embodiment illustrated in Figure 5, address element "BA 19" 591 can be a die select bit and can represent a most or least significant bit in a logical unit address. Embodiments are not limited to the example illustrated in Figure 5 as addresses can have more or less information,

e.g., greater or fewer numbers of bits, and the information can correspond to memory locations other than columns, planes, and blocks.

[0042] In one or more embodiments, control circuitry that is coupled to control inputs shared by a number of logical units can be configured to mask at least a portion of an address associated with one or more commands such that the one or more commands control two or more of the number of logical units. For example, two logical units (a first logical unit and a second logical unit) could have addresses differing only by one bit, e.g., a die select bit 591 or logical unit ID bit. In such an instance, the bit 591 can be masked such that a command addressed to either the first or the second logical unit would control both the first and the second logical units.

[0043] For example, consider a first logical unit having "00 11" as a portion of a logical unit address and a second logical unit having "101 1" as a portion of a logical unit address. The first bit could be masked by a number of different masking operations, e.g., logical operations. For instance, the portion of the address could be operated on logically by applying a bitwise AND with "01 11" such that the first bit would return a "0" for any entry in that field, e.g., bit 591. In this example, the result of the AND operation would be "001 1" for either the portion of the first logical unit address or the portion of the second logical unit address. Another example can include a bitwise OR operation with "1000" such that the bit 591 would be a "1". Such an operation would yield "101 1" for both the portion of the first logical unit address and the portion of the second logical unit address. The masked portion of the address can at least partially identify at least one of the number of logical units that are to be controlled by the masked address. In this example, either the first "0" or "1" is masked. That "0" or "1" can at least partially identify the first or second logical unit. That is, the bit to be masked can represent the only difference between the addresses of the first and second logical units.

[0044] Control circuitry can include additional gating, e.g., one or more AND gates, OR gates, or combinations of various logical gates to perform such masking operations. Additional gating can be located in control circuitry including command decoders, address registers, command registers, or other locations in the memory device depending on the particular implementation. Other masking operations are possible, as will be understood by one of ordinary

skill in the art having read and understood the present disclosure. Additionally, the address can be modified, e.g., by adding or subtracting a value therefrom such that it corresponds to two or more logical units. As described above, such a masking operation can be optionally enabled based on, for example, an operating parameter being set to a particular value. In one or more embodiments, only one bit of one or more addresses is masked. Regardless of how many bits are used to identify a particular logical unit, masking only one bit can provide control over two logical units. For embodiments where simultaneous control is desired over more than two logical units, more than one bit can be masked.

[0045] In one or more embodiments, control circuitry that is coupled to control inputs shared by a number of logical units can be configured to send one or more commands to two or more logical units without regard to at least a portion of an address associated with one or more commands such that the one or more commands control the two or more logical units. For example, two logical units (a first logical unit and a second logical unit) could have addresses differing only by one bit, e.g., a die select bit 591 or logical unit ID bit. In such an instance, control circuitry can send a command to both the first and the second logical units regardless of the status of a portion of the address, e.g., whether bit 591 is a 0 or a 1. That is, the control circuitry can "ignore" at least a portion of the address, e.g., bit 591.

[0046] In one or more embodiments, the number of logical units controlled by one or more commands can share a remainder of an address identifying each logical unit, other than the portion of the address that is masked, e.g., ignored. Thus, the portion of the address that is masked and the remainder of the address can identify one of the logical units controlled by the one or more commands. That is, the logical units that are controlled by the one or more commands can share a substantial portion of the address corresponding to each logical unit. For example, in table 590, 25 bits are used, e.g., not LOW, for the address. If two logical units are controlled by masking bit 591, then the two logical units share 24 out of the 25 bits that are included in their addresses. Embodiments are not limited to memory device operations that mask only one bit of an address corresponding to a logical unit.

[0047] Figure 6 illustrates a block diagram of a solid state drive 620 in accordance with one or more embodiments of the present disclosure. The

embodiment of Figure 6 illustrates the components and architecture of one embodiment of a solid state drive 620. In the embodiment illustrated in Figure 6, the solid state drive 620 includes a controller 601, an interface 603, and solid state memory arrays 630-1, . . . , 630-N. In one or more embodiments, the solid state drive 620 can include a housing to enclose the solid state drive 620, though such housing may not be included in some embodiments.

[0048] The interface 603 can be used to communicate information between the solid state drive 620 and another device such as a computing device. For example, when the solid state drive 620 is used for data storage in a computing device, as illustrated in Figure 3, the interface 603 can be a serial advanced technology attachment (SATA), among others.

[0049] The controller 601 can communicate with the solid state memory arrays 630-1, . . . , 630-N to sense, program, and erase data, among other operations, on the solid state memory arrays 630-1, . . . , 630-N. Controller 601 can have circuitry that may be one or more integrated circuits and/or discrete components. For one or more embodiments, the circuitry in controller 601 may include control circuitry for controlling access across a number of memory arrays and/or for providing a translation layer between an external host and the solid state drive 620. Thus, a memory controller could selectively couple an I/O connection (not shown in Figure 6) of a memory array to receive the appropriate signal at the appropriate I/O connection at the appropriate time. Similarly, the communication protocol between a host and the solid state drive 620 may be different than what is required for access of a memory array, e.g., memory array 630-1. Memory controller 601 could then translate the command sequences received from a host into the appropriate command sequences to achieve the desired access to a memory array. Such translation may further include changes in signal voltage levels in addition to command sequences.

[0050] The circuitry of controller 601 may further include functionality unrelated to control of a memory array, e.g., array 630-1, such as logic functions as might be performed by an ASIC. Also, the circuitry of controller 601 may include circuitry to restrict sensing or programming access to the solid state drive 620, such as password protection, biometrics or the like. The circuitry of controller 601 may include circuitry to indicate a status of the solid state drive 620. For example, the circuitry of controller 601 may include functionality to

determine whether power is being supplied to the solid state drive 620 and whether the solid state drive 620 is currently being accessed, and to display an indication of its status, such as a solid light while powered and a flashing light while being accessed. The circuitry of controller 601 may further include passive devices, such as decoupling capacitors to help regulate power requirements within the solid state drive 620.

[0051] Embodiments of the present disclosure can include a number of solid state memory arrays 630-1, . . . , 630-N. The solid state memory arrays 630-1, . . . , 630-N can be various types of volatile and/or non-volatile memory arrays (e.g., Flash or DRAM arrays, among others). Memory arrays 630-1, . . . , 630-N can include a number of memory cells that can be grouped in logical units such as a number of individual or stacked dice.

[0052] In one or more embodiments, the solid state drive can implement wear leveling to control the wear rate on the memory arrays 630-1, . . . , 630-N. As one of ordinary skill in the art will appreciate, wear leveling can increase the life of a solid state memory array since a solid state memory array can experience failure after a number of program and/or erase cycles.

[0053] In various embodiments, wear leveling can include dynamic wear leveling to minimize the amount of valid blocks moved to reclaim a block. Dynamic wear leveling can include a technique called garbage collection in which blocks with a number of invalid pages (i.e., pages with data that has been re-written to a different page and/or is no longer needed on the invalid pages) are reclaimed by erasing the block. Static wear leveling includes writing static data to blocks that have high erase counts to prolong the life of the block.

[0054] In some embodiments, a number of blocks can be designated as spare blocks to reduce the amount of write amplification associated with writing data in the memory array. A spare block can be a block in a memory array that can be designated as a block where data can not be written. Write amplification is a process that occurs when writing data to solid state memory arrays. When randomly writing data in a memory array, the memory array scans for free space in the array. Free space in a memory array can be individual cells, pages, and/or blocks of memory cells that are not programmed. If there is enough free space to write the data, then the data is written to the free space in the memory array. If there is not enough free space in one location, the data in the memory array is

rearranged by erasing, moving, and rewriting the data that is already present in the memory array to a new location leaving free space for the new data that is to be written in the memory array. The rearranging of old data in the memory array is called write amplification because the amount of writing the memory arrays has to do in order to write new data is amplified based upon the amount of free space in the memory array and the size of the new data that is to be written on the memory array. Write amplification can be reduced by increasing the amount of space on a memory array that is designated as free space (i.e., where static data will not be written), thus allowing for less amplification of the amount of data that has to be written because less data will have to be rearranged.

[0055] In various embodiments, host and/or user traffic and/or program/erase cycles performed by the solid state drive can be monitored, in addition to wear leveling in the solid state drive, to improve performance of the solid state drive. Host and/or user traffic requests can be made by the processor through the controller to read data and/or erase/write data on the solid state drive. Program and/or erase cycles can be monitored to determine the wear rate and life expectancy of the blocks and/or pages in the solid state memory arrays, as a solid state memory arrays can only be erased and written to a finite number of time. Host and/or user traffic trends can be monitored and altered to allow the drive to perform for a desired operational life (e.g., a time period such as hours, days, weeks, years, etc). The solid state drive can monitor and limit the number of program and/or erase cycles performed by the solid state drive, such as to ensure a desired operational life of the drive. The solid state drive can also monitor the number of program and/or erase cycles performed over a particular time period to determine how to calculate the allowable program and/or erase cycle rate for the drive given the a number of space blocks and the desired operational life of the drive.

[0056] In addition, in some embodiments, the number of spare blocks in the solid state memory arrays of the solid state drive can be controlled to ensure operability over a desired operational life for a desired number of write IOPs per unit time. The percentage of spare blocks can be optimized for the type of data that is being programmed on the solid state drive. A solid state drive that has static data (i.e., data that is stored on the drive for long periods of time without be erased and/or rewritten) can have a lower percentage of spare blocks because

there is less need to reclaim blocks in the drive due to the static nature of the data having fewer program and/or erase cycles. In a solid state drive that has dynamic data (i.e., data that is programmed and/or erased more frequently) a higher percentage of spare blocks can be used to reduce the write amplification associated with having to reclaim blocks to perform program and/or erase cycles in the memory array.

[0057] Figure 7 is a functional block diagram of a memory module 793 having at least one memory device operated in accordance with one or more embodiments of the present disclosure. Memory module 793 is illustrated as a memory card, although the concepts discussed with reference to memory module 793 are applicable to other types of removable or portable memory (e.g., USB flash drives and/or solid-state drives) and are intended to be within the scope of "memory module" as used herein. In addition, although one example form factor is depicted in Figure 7, these concepts are applicable to other form factors as well.

[0058] In one or more embodiments, memory module 793 will include a housing 794 (as depicted) to enclose one or more memory devices 720, though such a housing is not essential to all devices or device applications. At least one memory device 720 includes an array of non-volatile multilevel memory cells. Where present, the housing 794 includes one or more contacts 796 for communication with a host device. Examples of host devices include digital cameras, digital recording and playback devices, PDAs, personal computers, memory card readers, interface hubs and the like. For one or more embodiments, the contacts 796 are in the form of a standardized interface. For example, with a USB flash drive, the contacts 796 might be in the form of a USB Type-A male connector. For one or more embodiments, the contacts 796 are in the form of a semi-proprietary interface, such as might be found on CompactFlash™ memory cards licensed by SanDisk Corporation, Memory Stick™ memory cards licensed by Sony Corporation, SD Secure Digital™ memory cards licensed by Toshiba Corporation and the like. In general, however, contacts 796 provide an interface for passing control, address and/or data signals between the memory module 793 and a host having compatible receptors for the contacts 796.

[0059] The memory module 793 may optionally include additional circuitry 797, which may be one or more integrated circuits and/or discrete components. For one or more embodiments, the additional circuitry 797 may include control circuitry, such as a memory controller, for controlling access across multiple memory devices 720 and/or for providing a translation layer between an external host and a memory device 720. For example, there may not be a one-to-one correspondence between the number of contacts 796 and a number of connections to the one or more memory devices 720. Thus, a memory controller could selectively couple an I/O connection (not shown in Figure 7) of a memory device 720 to receive the appropriate signal at the appropriate I/O connection at the appropriate time or to provide the appropriate signal at the appropriate contact 796 at the appropriate time. Similarly, the communication protocol between a host and the memory module 793 may be different than what is used for access of a memory device 720. A memory controller could then translate the command sequences received from a host into the appropriate command sequences to achieve the desired access to the memory device 720. Such translation may further include changes in signal voltage levels in addition to command sequences.

[0060] The additional circuitry 797 may further include functionality unrelated to control of a memory device 720 such as logic functions as might be performed by an ASIC. Also, the additional circuitry 797 may include circuitry to restrict read or write access to the memory module 793, such as password protection, biometrics or the like. The additional circuitry 797 may include circuitry to indicate a status of the memory module 793. For example, the additional circuitry 797 may include functionality to determine whether power is being supplied to the memory module 793 and whether the memory module 793 is currently being accessed, and to display an indication of its status, such as a solid light while powered and a flashing light while being accessed. The additional circuitry 797 may further include passive devices, such as decoupling capacitors to help regulate power requirements within the memory module 793.

Conclusion

[0061] The present disclosure includes methods and devices for logical unit operation. One device embodiment includes a number of logical units,

wherein each of the number of logical units has a unique address. The device includes control circuitry coupled to the number of logical units and configured optionally to control more than one of the number of logical units with one of a number of commands and one address.

[0062] Although specific embodiments have been illustrated and described herein, those of ordinary skill in the art will appreciate that an arrangement calculated to achieve the same results can be substituted for the specific embodiments shown. This disclosure is intended to cover adaptations or variations of one or more embodiments of the present disclosure. It is to be understood that the above description has been made in an illustrative fashion, and not a restrictive one. Combination of the above embodiments, and other embodiments not specifically described herein will be apparent to those of skill in the art upon reviewing the above description. The scope of the one or more embodiments of the present disclosure includes other applications in which the above structures and methods are used. Therefore, the scope of one or more embodiments of the present disclosure should be determined with reference to the appended claims, along with the full range of equivalents to which such claims are entitled.

[0063] In the foregoing Detailed Description, some features are grouped together in a single embodiment for the purpose of streamlining the disclosure. This method of disclosure is not to be interpreted as reflecting an intention that the disclosed embodiments of the present disclosure have to use more features than are expressly recited in each claim. Rather, as the following claims reflect, inventive subject matter lies in less than all features of a single disclosed embodiment. Thus, the following claims are hereby incorporated into the Detailed Description, with each claim standing on its own as a separate embodiment.

What is Claimed is:

1. A memory device, comprising:
a number of logical units, wherein each of the number of logical units has a unique address; and
control circuitry coupled to the number of logical units and configured optionally to control two or more of the number of logical units with one of a number of commands and one address.
2. The device of claim 1, wherein the number of commands include programming commands and erasing commands.
3. The device of claim 2, wherein the number of commands include move commands, cached commands, and multiple plane commands.
4. The device of claim 1, wherein the number of commands do not include reading commands.
5. The device of claim 4, wherein the number of commands do not include status commands, get feature commands, or set feature commands.
6. The device of claim 1, wherein each of the two or more of the number of logical units are selected from the group of logical units including:
one or more dice of memory cells; and
more than one stacked dice of memory cells.
7. The device of any one of claims 1-6, wherein:
the device includes input/output (I/O) circuitry coupled to the number of logical units and to the control circuitry; and
the control circuitry is configured to transfer data associated with the one of the number of commands across the I/O circuitry only once to the two or more of the number of logical units.

8. A memory device, comprising:
 - a number of logical units, wherein at least two of the number of logical units share one or more control inputs;
 - control circuitry coupled to the one or more control inputs and configured optionally to mask at least a portion of an address associated with one or more commands such that the one or more commands control two or more of the number of logical units.
9. The device of claim 8, wherein the portion of the address includes one or more die select bits.
10. The device of claim 8, wherein the portion of the address includes only one bit.
11. The device of claim 8, wherein the portion of the address at least partially identifies at least one of the two or more of the number of logical units.
12. The device of claim 8, wherein the control circuitry includes one or more logic gates configured to mask the portion of the address.
13. The device of claim 8, wherein the control circuitry is configured to transfer the one or more commands across input/output (I/O) circuitry and subsequently to transfer the masked address across the I/O circuitry.
14. The device of any one of claims 8-13, wherein the control circuitry is configured optionally to mask at least the portion of the address based on an operating parameter programmed in the device.
15. A memory device, comprising:
 - a number of logical units, wherein at least two of the number of logical units share one or more control inputs;
 - control circuitry coupled to the one or more control inputs and configured optionally to send one or more commands to two or more of the number of

logical units without regard to at least a portion of an address associated with the one or more commands.

16. The device of claim 15, wherein the two or more of the number of logical units share a remainder of the address.

17. The device of claim 16, wherein the portion and the remainder together identify one of the two or more of the number of logical units.

18. The device of claim 15, wherein:
the address identifies one of the two or more of the number of logical units; and
the control circuitry is configured to send the one or more commands to the two or more logical units.

19. The device of claim 15, wherein each logical unit includes one or more semiconductor dies, wherein each semiconductor die includes one or more arrays of nonvolatile memory cells.

20. The device of claim 15, wherein the one or more control inputs include address latch enable (ALE), chip enable (CE), read enable (RE), ready/busy (R/B), write protect (WP), and input/output (IO) connections.

21. The device of any one of claims 15-20, wherein the one or more commands simultaneously control two or more of the number of logical units.

22. A method for operating a memory device, comprising:
receiving a command and an address corresponding to one logical unit;
and
controlling more than the one logical unit according to the command when an operating parameter is set to a particular value.

23. The method of claim 22, including setting the operating parameter to the particular value during initialization of the memory device.

24. The method of any one of claims 22-23, wherein:
receiving the command includes receiving a programming command; and
the method includes transferring data associated with the programming
command across input/output circuitry only one time for programming to the
more than one logical units.
25. The method of any one of claims 22-23, wherein receiving the command
includes receiving at least one command selected from the group of commands
including:
a programming command; and
an erasing command.
26. The method of claim 25, wherein receiving the command does not
include receiving a reading command.
27. The method of claim 22, wherein controlling more than one logical unit
includes controlling more than one logical unit, each having an address that
shares a substantial portion of the address corresponding to the one logical unit.
28. The method of any one of claims 22-23, wherein receiving the command
and the address includes receiving the command and the address by control
circuitry of the memory device from a host.
29. A method for operating a memory device, comprising:
masking at least a portion of an address when an operating parameter is
set to a particular value, wherein the address is associated with a command and
corresponds to one logical unit; and
controlling two or more logical units according to the command, wherein
the two or more logical units correspond to the masked address.
30. The method of claim 29, wherein masking at least the portion of the
address includes performing a logical operation on the address.

31. The method of claim 30, wherein performing the logical operation includes performing at least one selected from the group of logical operations including:
- a bitwise AND operation; and
 - a bitwise OR operation.
32. The method of claim 29, wherein masking at least the portion of the address includes modifying at least the portion of the address such that it corresponds to the two or more logical units.
33. The method of any one of claims 29-32, wherein controlling the two or more logical units includes simultaneously controlling the one logical unit and at least one other logical unit.
34. A method for operating a memory device, comprising:
- setting an operating parameter to a first value;
 - receiving a first command and a first address corresponding to a first logical unit;
 - controlling the first logical unit and at least one other logical unit according to the first command;
 - setting the operating parameter to a second value subsequent to setting the operating parameter to the first value;
 - receiving a second command and the first address corresponding to the first logical unit; and
 - controlling only the first logical unit according to the second command.
35. The method of claim 34, wherein setting the operating parameter includes programming data to a portion of the memory device reserved for storing one or more operating parameters.
36. The method of claim 34, wherein setting the operating parameter includes transferring a command, an address cycle, and a number of data cycles across input/output circuitry of the memory device during initialization of the memory device.

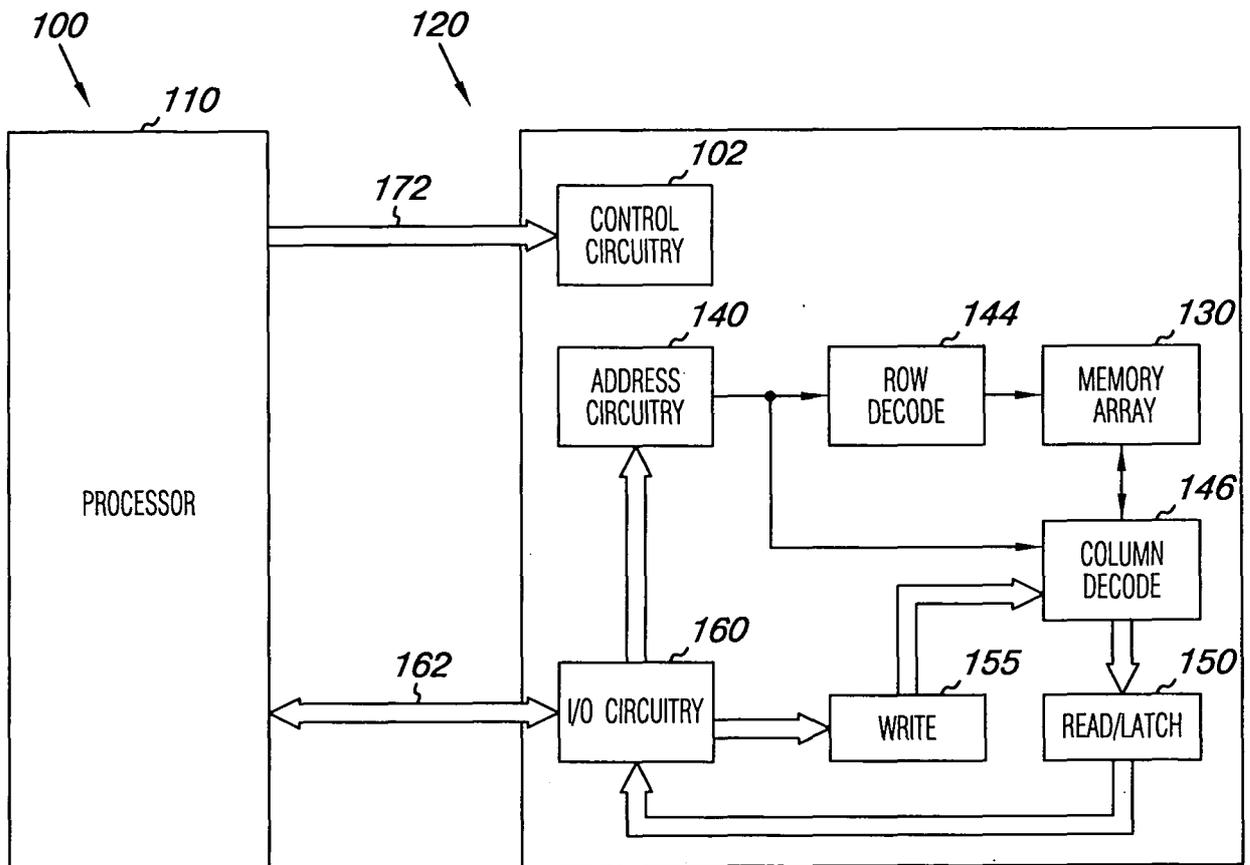


Fig. 1

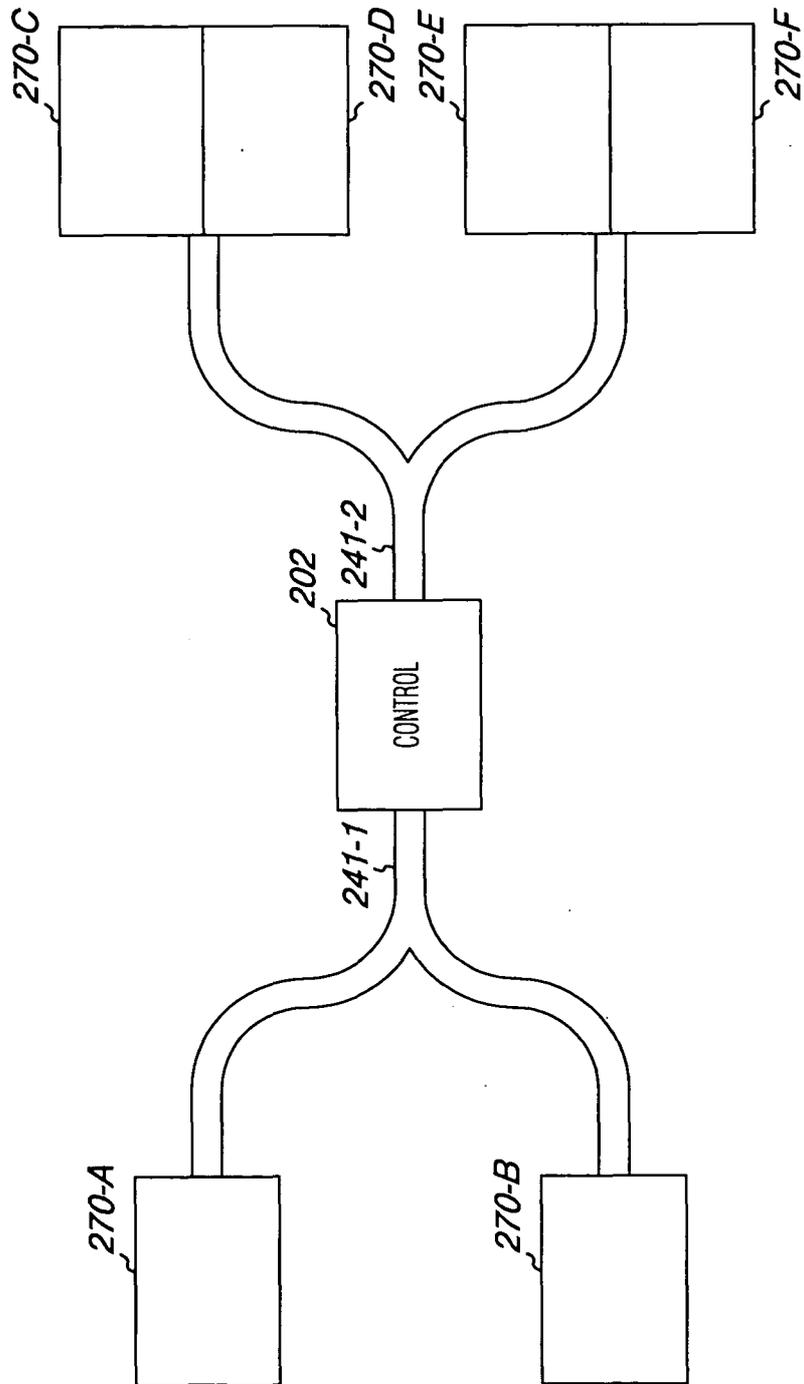


Fig. 2A

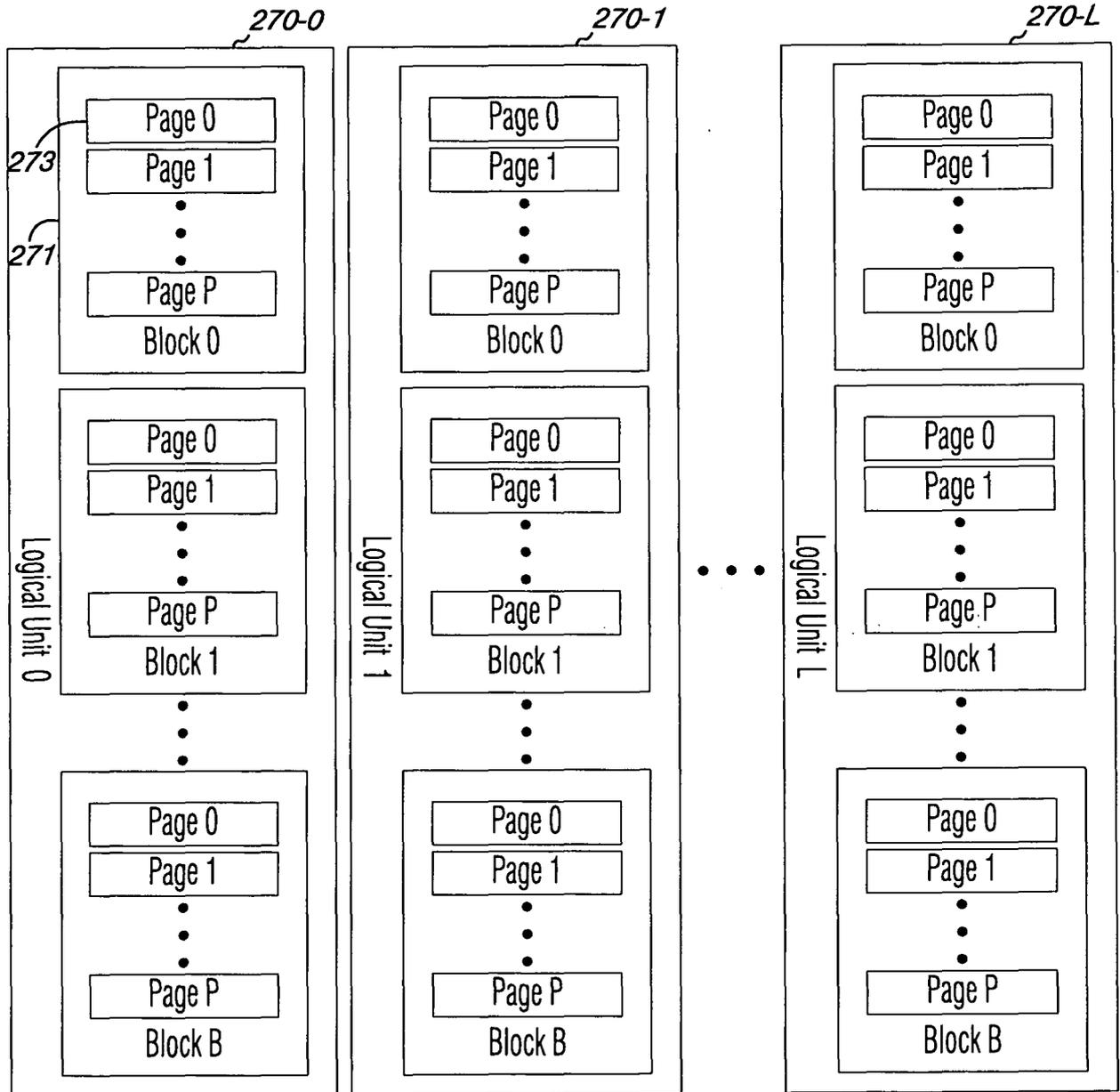


Fig. 2B

380 ↗

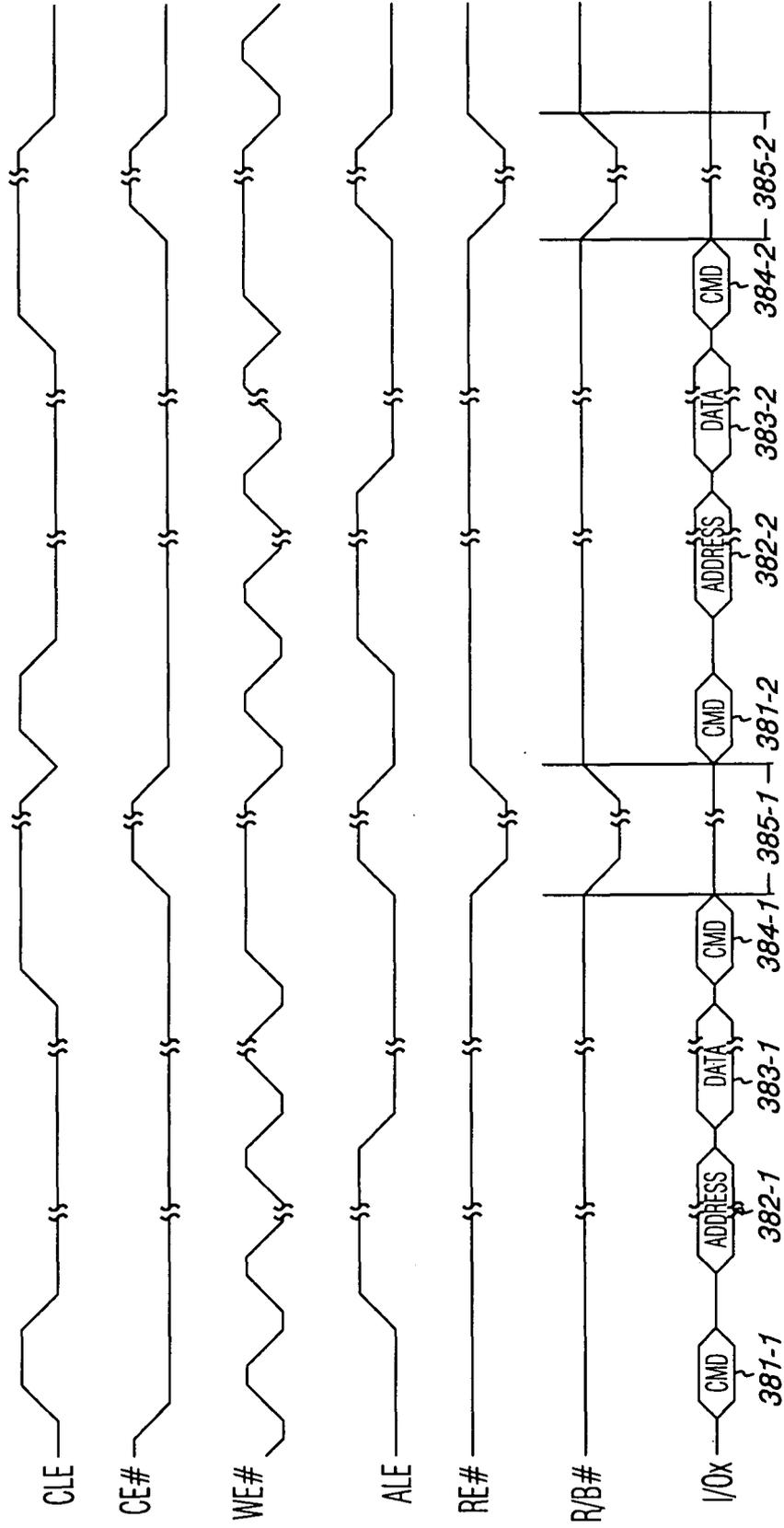


Fig. 3
PRIOR ART

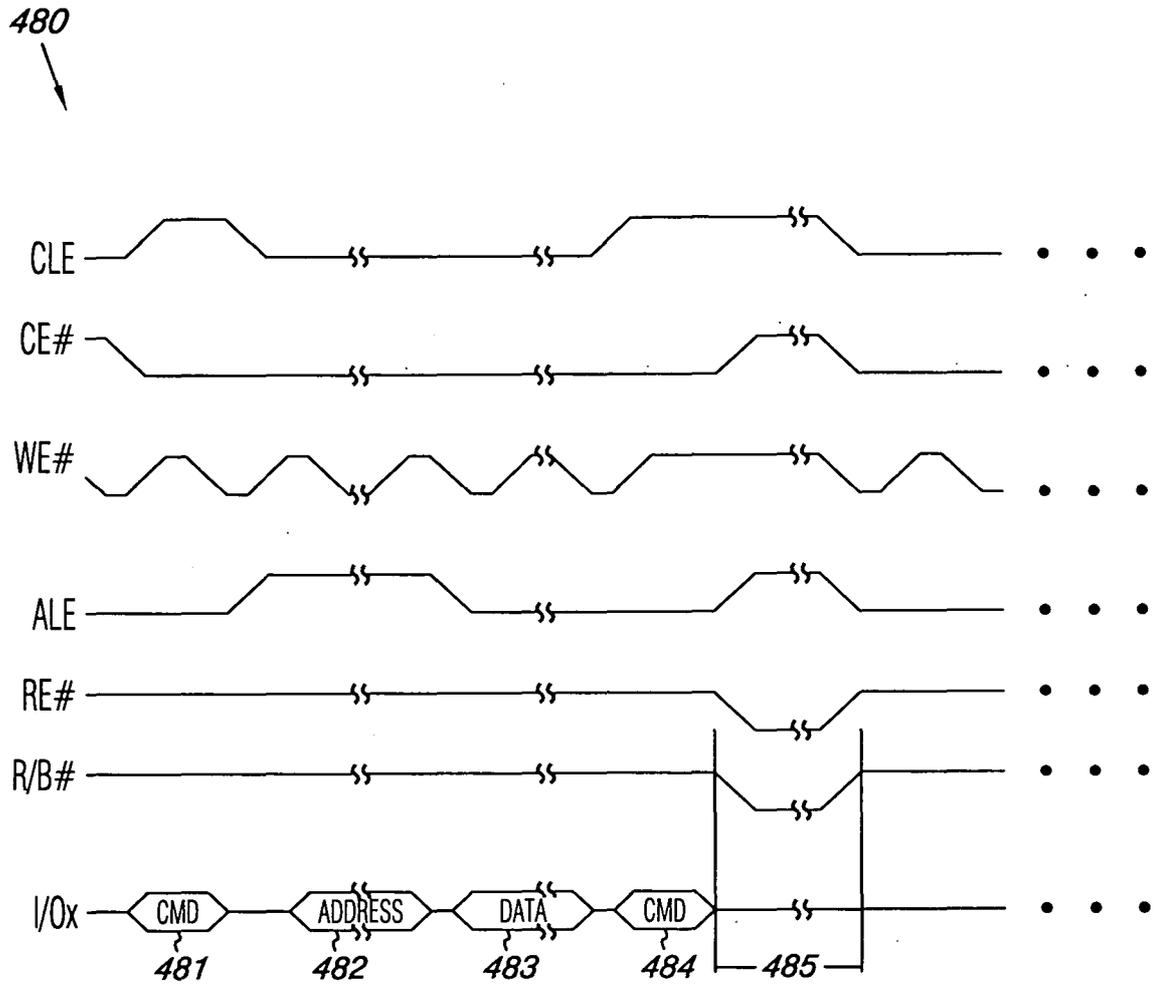


Fig. 4

590 

CYCLE	I/07	I/06	I/05	I/04	I/03	I/02	I/01	I/00
FIRST	CA7	CA6	CA5	CA4	CA3	CA2	CA1	CA0
SECOND	LOW	LOW	LOW	CA12	CA11	CA10	CA9	CA8
THIRD	BA7	PA6	PA5	PA4	PA3	PA2	PA1	PA0
FOURTH	BA15	BA14	BA13	BA12	BA11	BA10	BA9	BA8
FIFTH	LOW	LOW	LOW	LOW	BA19	BA18	BA17	BA16

591 

Fig. 5

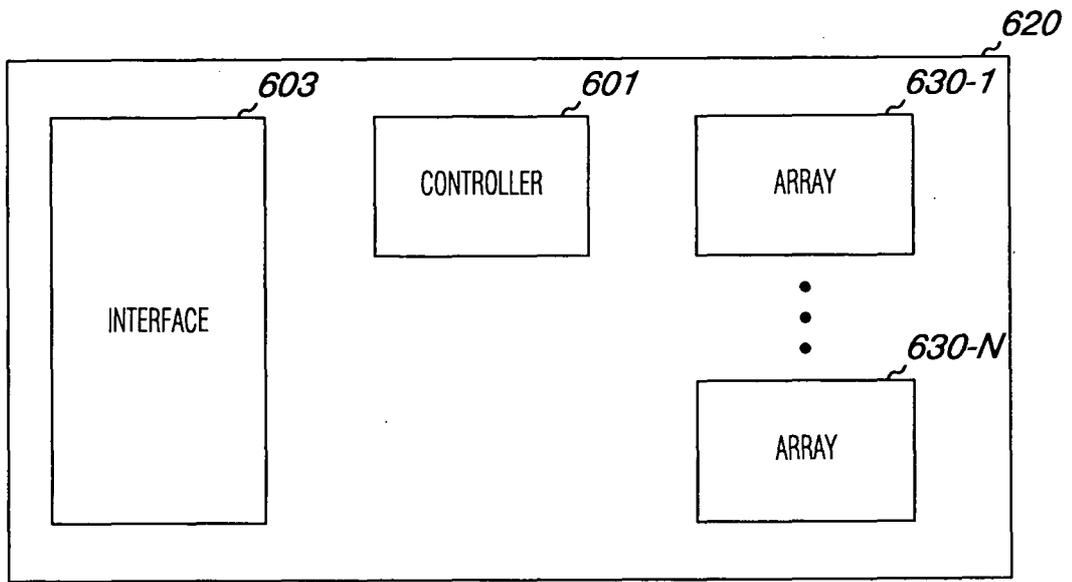


Fig. 6

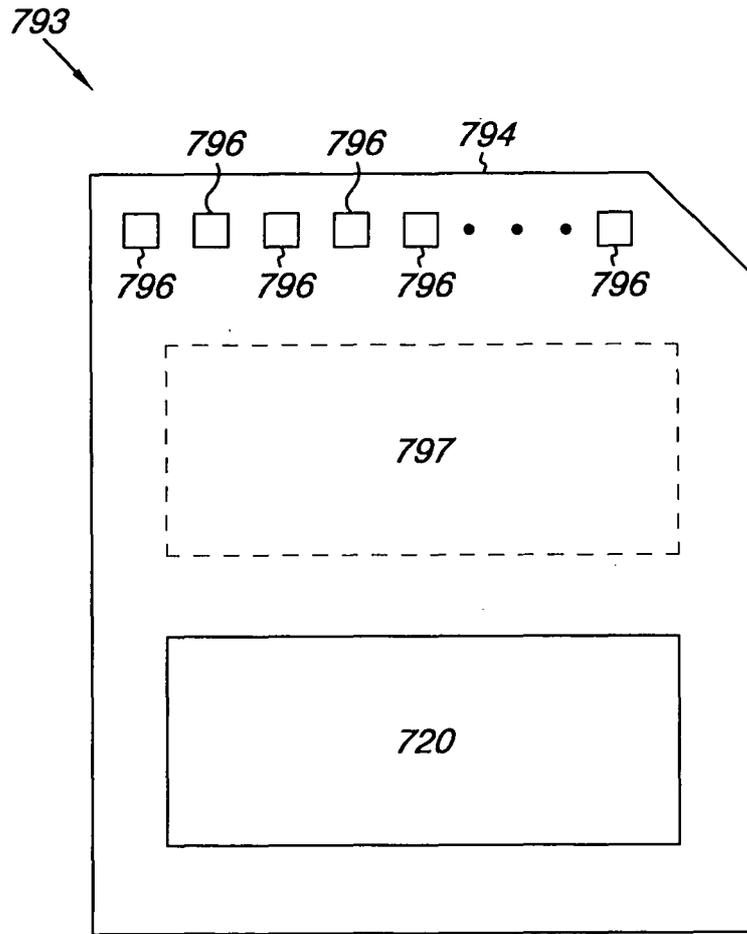


Fig. 7

A. CLASSIFICATION OF SUBJECT MATTER		
<i>G1C 8/00(2006.01)i, G1C 7/00(2006.01)i</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) IPC : G11C		
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: logical, unit, operation, control, address		
C. DOCUMENTS CONSIDERED TO BE RELEVANT		
Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	US 6813212 B2 (TAKAHASHI et al.) 02 November 2004 See the abstract, and the figure 1.	1-36
A	US 2003-0086295 A1 (HONDA et al.) 08 May 2003 See the abstract, and the figure 1.	1-36
A	US 2004-0095838 A1 (LI) 20 May 2004 See the abstract, and the figure 2.	1-36
<input type="checkbox"/> Further documents are listed in the continuation of Box C. <input checked="" type="checkbox"/> 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 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 13 APRIL 2010 (13.04.2010)		Date of mailing of the international search report 13 APRIL 2010 (13.04.2010)
Name and mailing address of the ISA/KR  Korean Intellectual Property Office Government Complex-Daejeon, 139 Seonsa-ro, Seo-gu, Daejeon 302-701, Republic of Korea Facsimile No. 82-42-472-7140		Authorized officer Yoon, Jin Hoon Telephone No. 82-42-481-5391 

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/US2009/005600

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 681 3212 B2	02. 11.2004	CN 1479924 A	03.03.2004
		CN 1479924 CO	21.06.2006
		EP 1351250 A1	08. 10.2003
		EP 1351250 A4	21. 12.2005
		JP 2002-184174 A	28.06.2002
		KR 10-2003-007 1783 A	06.09.2003
		TW 544676 A	01.08.2003
		TW 544676 B	01.08.2003
		US 2004-0027876 A1	12.02.2004
		WO 02-49037 A1	20.06.2002
		US 2003-0086295 A1	08.05.2003
EP 1052647 A3	13. 12.2000		
EP 1052647 B1	03. 12.2003		
JP 04047515 B2	13.02.2008		
JP 04393529 B2	23. 10.2009		
JP 2001-325795 A	22. 11.2001		
JP 2007-207425 A	16.08.2007		
JP 2007-226957 A	06.09.2007		
KR 10-0561567 B1	17.03.2006		
KR 10-0590 140 B1	15.06.2006		
KR 10-063606 1 B1	20. 10.2006		
US 2002-003 1038 A1	14.03.2002		
US 2004-02 18437 A1	04. 11.2004		
US 2005-0207247 A1	22.09.2005		
US 2006-02566 16 A1	16. 11.2006		
US 6377502 B1	23.04.2002		
US 6512693 B2	28.01 .2003		
US 6829194 B2	07. 12.2004		
US 6920057 B2	19.07.2005		
US 7126855 B2	24. 10.2006		
US 7345919 B2	18.03.2008		
US 2004-0095838 A1	20.05.2004	US 6850458 B2	01.02.2005