

(43) International Publication Date
8 March 2012 (08.03.2012)(10) International Publication Number
WO 2012/029092 A1(51) International Patent Classification:
G06F 3/06 (2006.01)(21) International Application Number:
PCT/JP2010/005403(22) International Filing Date:
2 September 2010 (02.09.2010)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant (for all designated States except US): **HITACHI, LTD.** [JP/JP]; 6-6, Marunouchi 1-chome, Chiyoda-ku, Tokyo, 1008280 (JP).

(72) Inventors; and

(75) Inventors/Applicants (for US only): **SANO, Takumi** [JP/JP]; C/O HITACHI, LTD., Disk Array Systems Division, 322-2, Nakazato, Odawara-shi, Kanagawa, 2500872 (JP). **KUROKAWA, Isamu** [JP/JP]; C/O HITACHI, LTD., Disk Array Systems Division, 322-2, Nakazato, Odawara-shi, Kanagawa, 2500872 (JP). **MORI, Akihiro** [JP/JP]; C/O HITACHI, LTD., Disk Array Systems Division, 322-2, Nakazato, Odawara-shi, Kanagawa, 2500872 (JP). **OGATA, Ran** [JP/JP]; C/O HITACHI, LTD., Disk Array Systems Division, 322-2, Nakazato, Odawara-shi, Kanagawa, 2500872 (JP). **GOTO, Yuya** [JP/JP]; C/O HI-

TACHI, LTD., Disk Array Systems Division, 322-2, Nakazato, Odawara-shi, Kanagawa, 2500872 (JP).

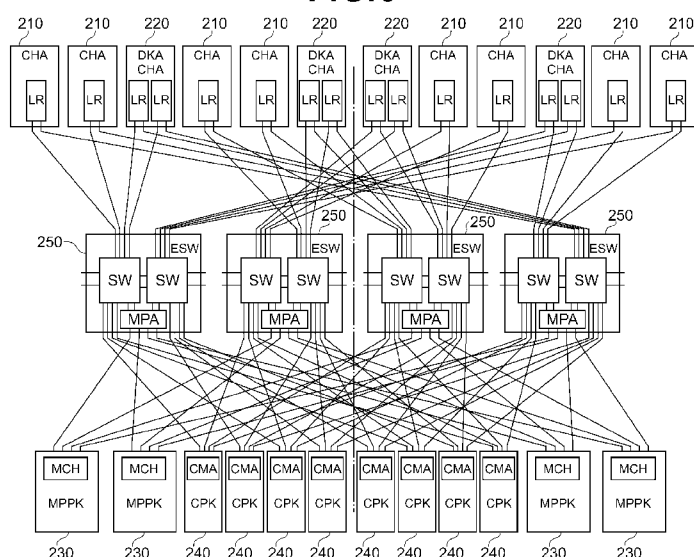
(74) Agents: **OHGA, Shinji** et al.; SUNNEXT INTERNATIONAL PATENT OFFICE, 3rd Floor, Shiodome Building, 1-2-20, Kaigan, Minato-ku, Tokyo, 1050022 (JP).

(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, 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, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, MD, 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, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

[Continued on next page]

(54) Title: STORAGE APPARATUS AND DATA ACCESS METHOD

FIG. 3

(57) Abstract: Improving response time performance by executing exclusive extent processing of a plurality of I/O requests is proposed. A storage apparatus, which includes a plurality of microprocessors, a first memory which stores counter values indicating the order of the I/Os to and from the storage areas, and a second memory which stores control information of the I/O requests for each of the I/O requests, is configured such that the microprocessors store, if there is one I/O request from the host apparatus, an I/O request range of the one I/O request in an area for storing control information of the one I/O request of the second memory, the microprocessors acquire the counter value from the first memory and store the counter value in the area for storing control information of the one I/O request of the second memory, and the microprocessors compare, if the counter value of the one I/O request is greater than the counter value of another I/O request, the I/O request range of the one I/O request

stored in the second memory with the I/O request range of the other I/O request.



Published:

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

Description

Title of Invention: STORAGE APPARATUS AND DATA ACCESS METHOD

Technical Field

[0001] The present invention relates to a storage apparatus and data access method, and is suitably applied to a storage apparatus and data access method which execute extent overlap determination processing for a plurality of I/O requests in parallel.

Background Art

[0002] A storage apparatus that is connected to host computers via a network comprises a plurality of magnetic disks, for example, as storage devices for storing data. The storage apparatus utilizes RAID (Redundant Arrays of Independent Disks) technology to establish redundancy of the storage areas of the plurality of storage devices and construct a RAID group. In addition, the storage apparatus provides the host computers with storage areas in the capacity required by the host computers and as logical volumes, from part of the RAID group.

[0003] Furthermore, if there is a data I/O request from the host computers, the storage apparatus executes exclusive extent processing, prior to executing the data I/O processing, so that there is no concurrent access to identical areas and to preserve the consistency of the access sequence. More specifically, exclusive extent processing is designed to prevent competing access to identical areas by determining whether or not there are I/O requests with overlapping extents (I/O request ranges) in the same logical volume among a plurality of I/O requests, and then if an overlap exists, awaiting I/O processing corresponding to the I/O requests, and if no overlap exists, continuing I/O processing corresponding to the I/O requests.

[0004] For example, PTL1 discloses an arrangement whereby, if an I/O request is sent to the storage apparatus, a control table of a logical volume targeted by the I/O request is locked and exclusive extent processing is executed.

Citation List

Patent Literature

[0005] [PTL 1] Japanese Laid-Open Patent Publication No. 2000-181773.

Summary of Invention

Technical Problem

[0006] However, in PTL1, if exclusive extent processing of an I/O request is being executed with respect to one logical volume, then exclusive extent processing of other I/O requests with respect to the logical volume enters a waiting state. As a result, due to an

increase in the multiplicity of I/O requests with respect to identical volumes, there is an increase in the I/O requests awaiting exclusive extent processing and therefore a deterioration in response time performance.

[0007] The present invention was conceived in view of the aforementioned points and proposes a storage apparatus and data access method with which exclusive extent processing of a plurality of I/O requests is executed in parallel in order to improve the response time performance.

Solution to Problem

[0008] In order to achieve this object, the present invention provides a storage apparatus which is connected via a network to a host apparatus which requests data I/Os, comprising a plurality of microprocessors which control data I/O to and from a plurality of storage devices on the basis of the data I/O requests from the host apparatus; a first memory which stores, for each storage area of a predetermined range provided by the plurality of storage devices, counter values indicating the order of the I/Os to and from the storage areas; and a second memory which stores control information of the I/O requests for each of the I/O requests, wherein the microprocessors store, if there is one I/O request from the host apparatus, an I/O request range of the one I/O request in an area for storing control information of the one I/O request of the second memory, acquire, from the first memory, the counter value of the storage area which is the target of the one I/O request and store the counter value in the area for storing control information of the one I/O request of the second memory, and compare, if the counter value of the one I/O request is greater than the counter value of another I/O request, the I/O request range of the one I/O request stored in the second memory with the I/O request range of the other I/O request.

[0009] With this configuration, when executing I/O processing, the microprocessor is able to provide a counter value to each of the logical volumes which are targets of the I/O processing and determine the access sequence for access to the logical volumes on the basis of these counter values. By providing the counter values, the counter values may be acquired and access by other I/O processing may be restricted only while the counter values are being incremented. It is therefore possible to execute in parallel exclusive extent processing of a plurality of I/O requests with respect to the same logical volume, thereby improving the response time performance.

Advantageous Effects of Invention

[0010] According to the present invention, the response time performance can be improved by executing, in parallel, exclusive extent processing of a plurality of I/O requests.

Brief Description of Drawings

[0011] [fig.1]A block diagram showing the whole configuration of a computer system

according to a first embodiment of the present invention.

[fig.2]A conceptual diagram showing an external configuration of a storage apparatus according to the first embodiment.

[fig.3]A block diagram showing a specific linear configuration of each logical substrate of a control module of the storage apparatus according to the first embodiment.

[fig.4]A block diagram showing an internal configuration of a microprocessor package according to the first embodiment.

[fig.5]A block diagram showing backend connections between storage media and disk adapters according to the first embodiment.

[fig.6]A conceptual view illustrating logical volumes according to the first embodiment.

[fig.7]A diagram showing an example of an ownership rights control table according to the first embodiment.

[fig.8]A conceptual view illustrating ownership rights according to the first embodiment.

[fig.9]A conceptual view providing an overview of conventional exclusive extent processing.

[fig.10]A conceptual view providing an overview of exclusive extent processing according to the first embodiment of the present invention.

[fig.11]A conceptual view showing an example of storage areas of the logical memory of the microprocessor package according to the first embodiment.

[fig.12]A conceptual view showing an example of storage points of counter values defined in the MP shared area according to the first embodiment.

[fig.13]A conceptual view showing an example of job control blocks defined in the MP shared area according to the first embodiment.

[fig.14]A table showing the details of extent management information which is stored in the job control blocks, according to the first embodiment.

[fig.15]A flowchart showing all I/O processing in the microprocessor of the storage apparatus according to the first embodiment.

[fig.16]A flowchart showing exclusive extent processing according to the first embodiment.

[fig.17]A flowchart showing details of exclusive extent processing according to the first embodiment.

[fig.18]A conceptual view of an example of an LDEV queue according to the first embodiment.

[fig.19]A conceptual view of an LDEV queue search system according to the first embodiment.

[fig.20]A chart showing an example of a search list according to the first embodiment.

[fig.20]A flowchart showing search list creation processing according to the first embodiment.

Description of Embodiments

[0012] An embodiment of the present invention will be described in detail hereinbelow with reference to the drawings.

[0013] (1) Hardware configuration of the computer system

As shown in Fig. 1, the computer system 1 according to this embodiment comprises a storage apparatus 10 and first and second hosts 20a, 20b (hereinafter the first and second hosts are referred to simply as the 'hosts 20'), a management server 30, and networks 40 and 50.

[0014] The hosts 20 are computer apparatuses comprising information processing resources such as a CPU (Central Processing Unit) and a memory, and are configured from a personal computer, workstation, mainframe or similar, for example. The CPU functions as an arithmetic processing unit and controls the operation of the hosts 20 in accordance with programs and computational parameters and the like which are stored in the memory. Furthermore, the hosts 20 comprise information input devices such as a keyboard, switch, pointing device, and/or microphone, and an information output device such as a display monitor and/or speaker.

[0015] Furthermore, the host 20 is connected to the storage apparatus 10 via the network 40. The network 40 is configured from a SAN (Storage Area Network) or the like, for example, and inter-device communications are executed in accordance with the Fibre Channel Protocol, for example. Furthermore, the network 40 may also be a LAN (Local Area Network), the Internet, a public line or a dedicated line or similar, for example. If the network 40 is a LAN, inter-device communications are executed in accordance with the TCP/IP (Transmission Control Protocol/Internet Protocol) protocol, for example.

[0016] The storage apparatus 10 is configured from information processing resources such as a CPU and memory and a plurality of storage media 15a, 15b, and 15c (may also be referred to hereinbelow simply as 'storage media 15'). Moreover, the storage apparatus 10 parses commands transmitted from the hosts 20 and executes data I/Os with respect to storage areas of the storage media 15.

[0017] The storage media 15 are configured, for example, from a plurality of hard disk drives (HDD: Hard Disk Drives) which are high-cost hard disk drives such as SSD (Solid State Disk), SCSI (Small Computer System Interface) disks or low-cost hard disk drives such as SATA (Serial AT Attachment) disks. The storage media 15 may also be referred to hereinbelow as 'hard disks 15'.

- [0018] The management server 30 is a computer apparatus comprising information processing resources such as a CPU and memory, and is configured from a personal computer, workstation, or mainframe or similar, for example. The CPU functions as an arithmetic processing unit and controls the operation of the management server 30 in accordance with programs and computational parameters and the like which are stored in the memory. Furthermore, the management server 30 comprises information input devices such as a keyboard, switch, pointing device, and/or microphone, or the like, and an information output device such as a monitor display and/or speaker, and manages the storage apparatus 10 in response to inputs from an operator or the like.
- [0019] Furthermore, the management server 30 is connected to the storage apparatus 10 via the network 50. The network 50 is configured from a SAN or LAN or the like, for example, similarly to the network 40.
- [0020] In addition, the program stored in the memory of the management server 30 is a program for managing the storage media 15 of the storage apparatus 10, and manages, for example, the configuration of the RAID group and the capacities of the logical volumes configured from part of the RAID group. The management server 30 changes the RAID configuration information and so forth by transmitting commands corresponding to user operations to the storage apparatus 10.
- [0021] (2) Storage apparatus hardware configuration
- Fig. 2 shows the external configuration of the storage apparatus 10. The storage apparatus 10 is configured from a module 100 (abbreviated as 'Module 0' in the drawings) and a module 101 (abbreviated as 'Module 1' in the drawings). Modules 100 and 101 have the same configuration and hence a detailed description of the configuration of module 100 will not be provided here.
- [0022] Module 100 comprises a control module 110 for controlling the overall operation of the storage apparatus 10 and a driving module 120 in which a plurality of disk drives are stored. As shown in Fig. 2, module 100 comprises three enclosures and a single enclosure is configured from three rack boxes. Among the nine rack boxes, the control module 110 is stored in one rack box (abbreviated as 'Logic BOX' in Fig. 2) and a driving module 120 is stored in the eight other rack boxes (abbreviated as 'HDD BOX' in Fig. 2).
- [0023] The rack boxes from which the control module 110 is configured are loaded with a plurality of logical substrates for controlling data I/Os between the hosts 20 and driver modules 120. Logical substrates which are loaded into the rack boxes which the control module 110 comprises include, for example, a CPU (Central Processing Unit) for controlling data I/Os, channel adapters for communicating with the hosts 20, and disk adapters for exercising control of data I/Os with respect to the disk drives stored in the driver modules 120.

- [0024] Fig. 3 shows a specific linear configuration of each logical substrate in the control module 110 of the storage apparatus 10. As shown in Fig. 3, the control module 110 is configured from a plurality of channel adapters (abbreviated as 'CHA' in the drawings) 210, a plurality of disk adapters (abbreviated as 'DKA' in the drawings) 220, a plurality of path switches 250 (abbreviated as 'ESW' in the drawings), a plurality of microprocessor packages (abbreviated as 'MPPK' in the drawings) 230, and a plurality of cache packages (abbreviated as 'CPK' in the drawings) 240.
- [0025] The channel adapters 210 are channel interface boards. The channel adapters 210 have a function for the data transfer of commands received from the hosts via local routers (abbreviated as 'LR' in Fig. 3). The disk adapters 220 are hard disk interface boards. The channel adapters 220 have a function for the data transfer of commands sent to the hard disks via local routers.
- [0026] The microprocessor packages 230 function as four-microprocessor CPUs. The microprocessor packages 230 include a memory and a processor which are connected via a memory controller hub (abbreviated as 'MCH' in the drawings) 232. The cache packages (abbreviated as 'CPK' in the drawings) 240 include a function for temporarily storing data. The cache packages 240 perform data transfers via cache memory adapters (abbreviated as 'CMA' in the drawings). The path switches 250 interconnect the microprocessor packages 230, the channel adapters 210, the disk adapters 220, and the cache packages 240 and include a function for distributing data by means of switches (abbreviated as 'SW' in the drawings).
- [0027] Data transfer in the control module 110 will now be described. First, data is transmitted from the hosts 20 via the channel adapters 210 and the data transmitted to the channel adapters 210 is stored. Notification that data has been transmitted from the hosts 20 is then sent to the microprocessor packages 230. Any of the microprocessors in the microprocessor packages 230 then acquire the data stored in the channel adapters 210. In addition, the microprocessors execute data reads/writes and so on based on instructions from the hosts 20 and store the execution results in the cache packages 240. The microprocessor then informs the channel adapters 210 that data read/write processing or the like has been executed. Furthermore, the channel adapters 210 then transmit the processing results stored in the cache packages 240 to the hosts 20.
- [0028] It is assumed in the foregoing that the data transmitted from the hosts 20 is stored in the channel adapters 210 but storage is not limited to this example. For example, the data transmitted from the hosts 20 may also be stored in shared memory (not shown) separate from the channel adapter 210. In this case, the microprocessors of the microprocessor packages 230 acquire the data stored in the shared memory and execute data reads/writes and so on based on instructions from the hosts 20.

- [0029] Fig. 4 shows an internal configuration of the microprocessor package 230. The microprocessor package (abbreviated as 'MPPK' in the drawings) 230 comprises a Logical Memory (LM) 231, a Memory Controller Hub (MCH) 232, a I/O Controller Hub (ICH) 233, a Flash Memory (FM) 234, a CPU 235, and a Front Side Bus (FSB) 236.
- [0030] The logical memory 231 is a local memory in a microprocessor package 230 and temporarily stores programs used in the execution of the CPU 235 and parameters and so on which suitably vary in its execution. The local memory 231 is mutually connected to a CPU 235 by means of a hub which is configured from a memory controller hub 232 and a bus which is configured from a front side bus 237, and so forth.
- [0031] The flash memory 234 is an involatile memory which stores programs and computation parameters and so on which are used by the CPU 235. The flash memory 234 is mutually connected to the CPU 235 by means of a hub which is configured from the I/O controller hub 233 and a bus which is configured from the front side bus 237, and so forth.
- [0032] The CPU 235 functions as an arithmetic processing device and a controller and controls the overall operation in the microprocessor packages 230 according to various programs. In addition, the CPU 235 is configured from four Microprocessors (MP) 236a, 236b, 236c, 236d (hereinafter referred to as 'microprocessors 236').
- [0033] Fig. 5 shows backend connections between the disk adapters 220 and storage medium 15. As shown in Fig. 5, the disk adapters 220 of the control module 110 are interconnected with the driver modules 120 via SAS controllers 221.
- [0034] The disk adapters 220 transmit data read/write commands to the driver modules 120 based on control of the microprocessors 236 in the microprocessor package 230. In addition, the disk adapters 220 transmit the data read/write commands via the plurality of SAS controllers 221. The SAS controllers 221 transmit commands to hard disks which are data read/write targets.
- [0035] Furthermore, as shown in Fig. 5, the driver module 120 comprises a plurality of hard disks 15. The plurality of hard disks 15 are arranged in equal numbers at the front (abbreviated as 'Front' in the drawings) and rear (abbreviated as 'Rear') of the enclosure. The hard disks 15 disposed at the front of the enclosure are connected in series by means of a plurality of expanders (abbreviated as 'Exp1' and 'Exp2' in the drawings). The expanders include a function for distributing data read/write commands transmitted from the disk adapters 220 to each of the hard disks. Furthermore, the hard disks 15 disposed at the rear of the enclosure are connected in series by means of a plurality of expanders in the same way as the hard disks 15 disposed at the front of the enclosure.

- [0036] As shown in Fig. 6, one or more logical volumes (abbreviated as 'LDEV' in the drawings) are configured in the storage area provided by the plurality of hard disks 15. In addition, the storage area provided by the hard disks 15 may also be managed as a pool that is supplied to the host 20. A pool may be a pool which is provided by the RAID group or a pool provided by Thin Provisioning, for example. For example, in the case of a pool that is provided by the RAID group, one RAID group is configured from a plurality of disks and one or a plurality of logical volumes are defined in the storage area provided by one RAID group. Furthermore, logical volumes which are provided by a plurality of RAID groups are managed as a single pool.
- [0037] In addition, in this embodiment, LDEV ownership rights are configured for each microprocessor package 230. The LDEV ownership rights are rights for accessing LDEV control information. If LDEV ownership rights have not been configured for each microprocessor package 230, I/O processing must be executed with respect to the target logical volume after restricting access to the logical volumes of other microprocessor packages. In this case, information indicating which microprocessor package 230 has accessed which logical volume is stored in the aforementioned shared memory.
- [0038] The microprocessors 236 may therefore need to access the shared memory when executing exclusive extent processing. However, the time for accessing the shared memory will then be from a few times to several hundred times the time required to access the memory in the microprocessor package 230. Moreover, because, when accessing data stored in the shared memory, access is made with the shared memory locked, if the microprocessors 236 of the plurality of microprocessor packages 230 share the shared memory, the processing time for accessing the shared memory is prolonged.
- [0039] Therefore, in this embodiment, by configuring LDEV ownership rights for each of the microprocessor packages 230, access to the shared memory can be eliminated in the data read/write processing in order to improve access performance. In addition, by configuring the LDEV ownership rights, the control information which is required in exclusive extent processing (described subsequently) can be managed only in an area which is shared by the microprocessors 236 in the microprocessor packages 230.
- [0040] For example, an ownership rights control table 700 for configuring ownership rights for each of the logical volumes can be stored in the memory of the storage apparatus 10 so that ownership rights can be configured for each of the microprocessor packages 230. As shown in Fig. 7, the ownership rights control table 700 is configured from an LDEV number field 701 and an ownership rights MPPK field number 702. The LDEV number field 701 stores numbers identifying logical volumes. The ownership rights MPPK number field 702 stores the numbers of microprocessor packages which include ownership rights corresponding to logical volume numbers. For example, MPPK0

possesses the ownership rights to LDEV 0 and 'LDEV 0' signifies the fact that this logical volume can only be accessed by the MPPK0. Similarly, MPPK1 possesses the ownership rights to the LDEV 1 and 'LDEV 1' signifies the fact that this logical volume can only be accessed by the MPPK1.

[0041] As shown in Fig. 8, each of the microprocessor packages 230 is only able to access the logical volumes configured in the ownership rights control table 700 and is unable to access logical volumes for which ownership rights have not been configured. Hence, according to this embodiment, there is no need to perform exclusion processing on different logical volumes because microprocessor packages 230 are configured so as to be capable of accessing each logical volume. However, four microprocessors 236 in a single microprocessor package 230 are capable of accessing the same logical volume simultaneously. It is therefore necessary to execute so-called exclusive extent processing so that there is no conflict of access to the same area by determining whether or not there is an extent overlap within the same logical volume.

[0042] Conventional exclusive extent processing and the exclusive extent processing according to this embodiment will be compared and described here. As shown in Fig. 9, conventionally if there are multiple I/O requests with respect to the same volume, a single I/O request possesses the control information of this volume and makes an extent overlap determination. For this reason, while a single I/O request is executing exclusive extent processing, other I/O requests enter a waiting state in which exclusive extent processing cannot be executed. As shown in Fig. 9, due to an increase in the multiplicity of I/O requests with respect to identical volumes, there is an increase in the extent overlap determination processing wait (C in the drawings) time and therefore a deterioration in the response time performance, which is problematic.

[0043] More specifically, conventionally, in exclusive extent processing it is determined whether or not there is a lock on I/O control information (SM) of access-target logical volumes. Subsequently, if there is no lock on the I/O control information of the access-target logical volumes, exclusive extent processing is executed in a state where there is a lock on the I/O control information. In addition, a state is assumed in which the I/O control information in logical volume units is locked while the exclusive extent processing is being executed. Other I/O processing therefore enters a waiting state until the lock is released while the exclusive extent processing is being executed.

[0044] Therefore, in this embodiment, if there are multiple I/O requests with respect to the same volume, a counter value is assigned to each I/O request so that exclusive extent processing of a plurality of I/O requests is executed in parallel. More specifically, I/O control information is only locked when a counter values assigned to logical volume units is acquired for each I/O request. As a result, the exclusive extent processing can be executed in parallel simply by performing exclusion processing so that two or more

I/O requests do not acquire counter values simultaneously. To this end, as shown in Fig. 10, the time taken by the control time is made as short as possible in order to allow the exclusive extent processing (time A in the drawings) to be executed in parallel.

[0045] (3) Storage apparatus software configuration

Fig. 11 shows the details of the logical memory 231 of the microprocessor package 230. The microprocessor package 230 contains one logical memory 231 and a single logical memory 231 is shared by four microprocessors 236. As shown in Fig. 11, the four microprocessors 236 use the storage area of the logical memory 231 by logically partitioning this storage area. In other words, the logical memory 231 is configured from an MP shared area 1101 which is shared and used by the four microprocessors 236, and MP dedicated areas 1102 which are used exclusively by each of the microprocessors 236. The logical memory 231 is logically partitioned and therefore each of the microprocessors 236 is able to access the dedicated areas of the other microprocessors.

[0046] Since the logical memory 231 is shared by four microprocessors 236, information in the logical memory 231 may be updated simultaneously by a plurality of the microprocessors 236. Therefore, in order to retain consistency of information in the logical memory 231, access to the logical memory 231 must be restricted if the plurality of microprocessors 236 update this information. Furthermore, the MP shared area 1101 is an area which stores information shared and used by a plurality of microprocessors 236, and which stores counter values indicating the I/O order, for example. Furthermore, the dedicated areas 1102 are areas which store information used when each of the microprocessors 236 accesses the logical volume (LDEV) to which it possesses ownership rights, and which store trace information indicating what kind of processing is performed by the microprocessors 236, for example.

[0047] Furthermore, Fig. 12 shows storage points of counter values defined in the MP shared area 1101. As shown in Fig. 12, the MP shared area 1101 stores counter values which are configured for logical volume (LDEV) units. The counter values fulfill the role of numbered tickets for accessing logical volumes with ownership rights and represent information which is used in I/O request exclusion control. The I/O requests acquire counter values of access-target logical volumes (LDEV) stored in the MP shared area 1101, and store these counter values in Job Control Blocks (JCB), which will be described subsequently.

[0048] The job control blocks will be described next with reference to Figs. 13 and 14. The job control blocks are areas which are used to control I/O processing or other jobs with respect to the logical volumes. A single job control block is assigned to a single job.

[0049] A job is configured from a command (CCW chain) which indicates an I/O operation from a host 20 which is connected to a channel adapter 210, and which is a single unit

of an I/O request. Examples of CCW chains include, for example, a command 'x63' to define extent ranges, a command 'x47' for searching for target records, and a command 'x06' for reading data, and so on.

[0050] Furthermore, the job control blocks store job-related information such as extent ranges, counter values, and I/O attributes such as read processing or write processing. In addition, as shown in Fig. 13, the job control blocks are defined in microprocessor units in the MP shared area 1101.

[0051] When executing jobs such as I/O processing, the microprocessors 236 each refer to the job control blocks of each job stored in the MP shared area 1101 and control job execution. The microprocessors 236 use the job control blocks which are allocated until job execution ends.

[0052] Furthermore, after the job ends, the job control blocks enter a free state in which they can be used by other job control blocks. Furthermore, a job control block which is in a usable state is used to control jobs which are executed subsequently. The job control blocks are used repeatedly by a plurality of jobs.

[0053] Furthermore, the job control blocks are each assigned an individual JCB number and the microprocessors use the JCB numbers to determine the job processing order and so on. The job control blocks can also be defined in 8K units for each of the microprocessors 236. That is, a 32K-job control block can be defined for a single microprocessor package 230.

[0054] Fig. 14 shows details of extent management information 1302 which is stored in the job control blocks. As shown in Fig. 14, examples of the extent management information 1302 include, for example, an extent processing controller, a counter value, a start position address, an end position address, and so on.

[0055] The extent processing controller is information for managing the status of the I/O processing, i.e. information such as an extent reservation bit, an exclusive extent waiting bit, an extent significant bit, a counter value significant bit, and so forth.

[0056] The extent reservation bit is information which is configured (bit ON) when it is determined, as a result of a job executing exclusive extent processing, that I/O processing can be continued. The exclusive extent waiting bit is configured when, as a result of a job executing exclusive extent processing, it is determined that I/O processing cannot be continued, that is, that I/O processing is in a waiting state.

[0057] Furthermore, the extent significant bit is configured in cases where the 'start position address' and 'end position address' in the job control blocks are valid values. In other words, when this bit is not configured, the 'start position address' and 'end position address' in the job control blocks are invalid and cannot be trusted.

[0058] Furthermore, the counter value significant bits are configured when the microprocessors 236 acquire counter values in the MP shared area 1101. That is, when these

bits are not configured, the 'counter values' stored in the job control blocks are invalid and cannot be trusted.

[0059] (4) Details of storage apparatus operation

Fig. 15 shows the flow of all the I/O processing in the microprocessor 236 of the storage apparatus 10. As shown in Fig. 15, the microprocessors 236 first execute I/O startup processing (S100). In specific terms, the microprocessors 236 perform poling at regular intervals of the memory in the microprocessor package 230 in order to detect whether or not I/O data has been transmitted from the hosts 20 via the channel adapters 210.

[0060] Furthermore, if it is detected in step S100 that data has been transmitted from the hosts 20, the microprocessors 236 execute exclusive extent processing to execute the I/O data (S200). The exclusive extent processing in step S200 will be described in detail subsequently.

[0061] The microprocessors 236 then execute I/O processing (I/O) based on the I/O data transmitted from the hosts 20 after executing exclusive extent processing in step S200 (S300). Specifically, the microprocessors 236 execute reading/writing of the data stored on the hard disks 15.

[0062] The microprocessors 236 then execute I/O end processing (S400). More specifically, the microprocessors 236 notify the channel adapters 210 that I/O processing has ended in order to inform the hosts 20 that I/O processing has been executed.

[0063] Fig. 16 is a flowchart showing details of exclusive extent processing of step 200 in Fig. 15. As shown in Fig. 16, the microprocessors 236 first store the access range (extent) for their own I/O processing (I/O) in the job control blocks (S202).

[0064] The job control blocks are areas assigned to each I/O processing and which store control information for I/O processing when the microprocessors 236 execute the I/O processing. Hence the microprocessors 236 store the access range for their own I/O processing in the job control block which corresponds to their own I/O processing. Furthermore, the access range is indicated by the start position address and end position address of the logical volume which is the access target.

[0065] The microprocessors 236 then turn ON the extent significant bit of the extent management information 1302 stored in their own job control blocks after storing their own access range in step S202 in the job control blocks (S204).

[0066] As mentioned hereinabove, the extent significant bit is a flag which is configured in cases where the 'start position address' and 'end position address' are valid values in the job control blocks, and is a flag which is configured when the microprocessors 236 store their own access range in the job control blocks. The microprocessors 236 execute exclusive extent processing if the extent significant bit of another I/O job control block is ON, that is, if the other I/O access range is valid.

- [0067] The microprocessors 236 then lock the counter value area of the access-target logical volume (LDEV) (S206). More specifically, the microprocessors 236 lock the area storing the access-target logical volume counter value, in the MP shared area 1101 of the logical memory 231 in the microprocessor package 230.
- [0068] The microprocessors 236 subsequently acquire the counter values of the access-target logical volumes in the MP shared area 1101 of the logical memory 231 in the microprocessor package 230, and store these counter values in the job control blocks corresponding to their own I/O processing (S208). The microprocessors 236 then add one to (increment by one) the counter values of the access-target logical volumes in the MP shared area 1101 of the logical memory 231 in the microprocessor package 230 (S210).
- [0069] The microprocessors 236 then release the lock on the counter value area of the access-target logical volumes in the MP shared area 1101 of the logical memory 231 in the access microprocessor package 230 (S212).
- [0070] The microprocessors 236 then turn ON the counter value significant bit of the extent management information 1302 stored in their own job control blocks (S214). Like the extent significant bits, the counter value significant bits are flags to which the microprocessors 236 refer when executing exclusive extent processing. The microprocessors 236 execute exclusive extent processing if the counter value significant bit of another I/O is ON, that is, if the counter value is valid.
- [0071] The microprocessors 236 then execute LDEV queue search processing (S216). The LDEV queue is an I/O wait queue in the microprocessor package 230, i.e. an LDEV unit queue. The LDEV queue is a wait queue which links job control blocks of I/O which access or are accessing the LDEV. The act of introducing data to a queue is known as 'enqueue' and the act of taking data from a queue is known as 'dequeue'. Processing to search for LDEV queues in step S216 will be described in detail subsequently.
- [0072] The microprocessors 236 then execute exclusive extent determination processing (S218). More specifically, exclusive extent determination processing is processing with which [the microprocessors 236] compare the extent range of job control blocks of the LDEV queue retrieved in step S216 with the extent range of their own I/O job control blocks.
- [0073] Details of the exclusive extent processing in step S218 will be described in detail subsequently. Fig. 17 shows details of exclusive extent processing. As shown in Fig. 17, the microprocessors 230 acquire extent management information for other I/O (S402). More specifically, the microprocessors 230 refer to a search list created by the aforementioned LDEV queue search processing and acquire the extent management information 1401 of the job control blocks which are targets of the exclusive extent

processing. The extent management information 1401 is information, for managing extent processing, which includes the counter values shown in Fig. 14.

[0074] The microprocessors 230 then refer to the extent significant bits of the extent management information 1401 of job control blocks of other I/O to determine whether or not the extents of the other I/O are valid (S404). In specific terms, the microprocessors 230 determine that an extent is valid if the extent significant bit is ON. If it is determined in step S404 that the extent of the other I/O is invalid, the microprocessors 230 execute the processing of step S412 and beyond.

[0075] If it is determined in step S404 that the extent of the other I/O is valid, the microprocessors 236 then refer to the counter value significant bits of the extent management information 1401 of job control blocks of other I/O to determine whether or not the extent of the other I/O are valid (S406). In specific terms, the microprocessors 236 determine that a counter value is valid if the counter value significant bit is ON. If it is determined in step S406 that the counter value of the other I/O is invalid, the microprocessors 236 execute the processing of step S412 and beyond.

[0076] If it is determined in step S406 that the extent of the other I/O is valid, the microprocessors 236 then compare their own I/O counter value with the counter value of the other I/O, and determine whether or not the counter value of its own I/O is greater than the counter value of the other I/O (S408). If the counter value of its own I/O is greater than the counter value of the other I/O, its own I/O follows on from the other I/O, and it is therefore necessary to determine whether or not there are overlapping extent ranges. If, on the other hand, the counter value of its own I/O is smaller than the counter value of the other I/O, the other I/O follows on from its own I/O. In this case, because its own I/O is processed in advance of the other I/O, it is not necessary to determine whether or not there are overlapping extent ranges.

[0077] If it is determined in step S408 that the counter value of the other I/O is smaller than the counter value of the other I/O, the microprocessors 236 execute the processing of step S412 and beyond. If, on the other hand, it is determined in step S408 that the counter value of its own I/O is greater than the counter value of the other I/O, the microprocessors 236 determine whether or not there is no overlap between the extent range of its own I/O with the extent range of the other I/O (S410). In specific terms, the microprocessors 236 compare ranges indicated by the start position address and the end position address of the extent management information of the job control blocks of their own I/O and other I/O in order to determine whether or not there is a degree of overlap between these ranges.

[0078] If it is determined in step S410 that there is no overlap between the extent range of its own I/O with the extent range of the other I/O, the microprocessors 236 determine whether or not its own I/O has been compared with all the other I/O (S412). In specific

terms, the microprocessors 236 determine whether or not the management information of the job control blocks of their own I/O has been compared with the management information of the job control blocks of all the other I/O listed in the search list shown in Fig. 19.

- [0079] Furthermore, if it is determined in step S410 that there is an overlap between the extent range of its own I/O with the extent range of the other I/O, the microprocessors 236 turn ON the exclusive extent waiting bit (S418). More specifically, the microprocessors 236 then turn ON the exclusive extent waiting bit of the extent management information of the job control blocks shown in Fig. 14. If the exclusive extent waiting bit is ON, the microprocessors 236 are unable to execute their own I/O processing and wait to execute their own I/O processing until the I/O processing of the other I/O with which an extent range overlap exists has finished.
- [0080] If it is then determined in step S412 that their own I/O has been compared with all the other I/O, the microprocessors 236 then turn OFF the exclusive extent waiting bit extent of the extent management information of their own job control blocks and turn ON the extent reservation bit (S414).
- [0081] The exclusive extent waiting bit and extent reservation bit of the extent management information have an opposing relationship. If the exclusive extent waiting bit is ON, this indicates that the microprocessors 236 are unable to continue their own I/O processing and are waiting for the end of the other I/O processing with which there is an extent range overlap. Furthermore, a case where the extent reservation bit is ON is configured in cases where the microprocessors 236 determine that their own I/O processing can be continued. Details of exclusive extent processing were described hereinabove.
- [0082] Subsequently, details of LDEV queue search processing will be provided next. As shown in Fig. 18, the LDEV queues are stored in the MP shared area 1101 which is shared and used by four microprocessors 236. The LDEV queue area 1702, which stores each of the microprocessor LDEV queues, stores the queues of the LDEV for which the microprocessor packages 230 have ownership rights.
- [0083] For example, if the job control blocks which access or are accessing LDEV 0 are JCB (9), JCB (3), and JCB (5), information is stored indicating that the queues of the MP 0 LDEV 0 are waiting queues linking the three job control blocks JCB (9), JCB (3), and JCB (5). When searching for the LDEV queues, a search is performed for all the job control blocks of the target LDEVs which are stored in the memory of four microprocessors, namely MP 0, MP 1, MP 3, and MP 4.
- [0084] Fig. 19 is a conceptual view to illustrate an LDEV queue search system. As mentioned earlier, LDEV queues are stored for each LDEV in the memory of each of the microprocessors 236. The microprocessors 236 search for the LDEV queues of the

processing target LDEV of their own I/O.

[0085] In specific terms, the microprocessors 236 search for their own LDEV queues and the LDEV queues of the other microprocessors in the same microprocessor package 230. For example, as shown in Fig. 18, when searching for the LDEV queue of LDEV 1, the microprocessors 236 search for the job control blocks of each of the LDEV queues in order working from the MRU (Most Recently Used) toward the LRU (Least Recently Used).

[0086] When I/O processing is initiated, the job control blocks of new I/O processing are enqueued starting with the MRU of the LDEV queues. Normally, when executing a search in order working from the LRU toward the MRU, leakage in the search during queue operation will likely occur. For example, when searching for the LDEV queues, the job control blocks in the search are dequeued and job control blocks of the same JCB number as the job control blocks dequeued by subsequent I/O processing are enqueued starting with the MRU. In this case, LDEV queue search leakage arises between the job control blocks being searched and the job control blocks which are dequeued starting with the MRU.

[0087] Suppose, for example, that in a certain LDEV queue, the queues are linked starting with the LRU in the order JCB (3), JCB (1), and then JCB (9), and that JCB (1) is dequeued when the microprocessors 236 are searching for the JCB (1). When JCB (1) is being used by other I/O processing, the JCB (1) is then enqueued starting with the MRU.

[0088] Here, the queues are linked starting with the LRU in the order JCB (3), JCB (9), and then JCB (1). At that point, the microprocessors 236 searching for the queues starting with the LRU end the search based on the fact that there is no linked queue following the MRU JCB (1) and the JCB (9) is leaked from the search target.

[0089] Therefore, in this embodiment, search leakage is prevented by making the search direction extend not from the LRU to the MRU but instead from the MRU to the LRU. However, in cases where the LDEV queue search processing is executed starting with the MRU, if queue operations such as dequeuing or enqueueing are generated in the job control blocks being searched for and if a queue returns to the MRU, the search must be executed once again starting with the MRU.

[0090] When the processing time for the search processing of a single job control block is extended by the microprocessors 236, the queue operation of the job control blocks will probably occur frequently and the search processing will take time. Therefore, before LDEV queue search processing is executed, a search list for a search starting with the MRU and ending with the LRU is created and search processing is executed by referring to the search list. As a result, the processing time for the search processing can be shortened while preventing search leakage.

- [0091] Furthermore, when a search list indicating the search order of job control blocks is created, the JCB numbers of the job control blocks linked to the LDEV queue of each of the microprocessors 236 and the JCB generation numbers are acquired. The job control blocks are used as areas for storing control information of I/O processing by the plurality of I/O processes. The microprocessors 236 identify which job control blocks are used by which I/O processing from the JCB numbers and generation numbers of the job control blocks.
- [0092] Fig. 20 shows an example of a search list which is created before search processing is executed. The search list 1900 is a list to which the microprocessors 236 refer when executing LDEV queue search processing and which is configured from an MP number (MP No) field 1901, a JCB number (JCB No) field 1902, and a generation number (generation No) field 1903.
- [0093] The MP number field 1901 stores numbers identifying microprocessors 236 in those microprocessor packages 230 to which the microprocessors 236 being searched belong. The JCB number field 1902 stores JCB numbers of the job control blocks linked to the LDEV queue. The generation number field 1903 stores the generation numbers of each of the job control blocks.
- [0094] The microprocessors 236 acquire the JCB numbers stored in the search list and execute exclusive extent processing in the order of the search list. Although I/O processing will likely also be initiated from the point where the microprocessors 236 create the search list, it is not clear that the I/O processing follows its own I/O processing and hence need not be made a search target.
- [0095] Fig. 21 shows the flow of search list creation processing before the microprocessors 236 execute search processing. The search list creation processing targets the LDEV queues of four microprocessors 236 in the microprocessor package 230. The microprocessors 236 create a search list by taking, as search targets, not only the job control blocks of their own microprocessors 236 but also the job control blocks of the other microprocessors.
- [0096] In reality, the microprocessors 236 replace the variable 'number of JCBs' with '0' (S302), and the variable 'number of MPs' with '0' (S304). The microprocessors 236 then add '1' to the variable 'number of MPs' (S306), and add '1' to the variable 'number of JCBs' (S308).
- [0097] In the subsequent determination processing, by determining whether or not the variable 'number of MPs' is equal to or more than the number '4' in the microprocessor package 230 and determining whether or not the variable 'number of JCBs' is equal to or more than '8K', a search list can be created by taking, as search targets, all the job control blocks in the microprocessor package 230.
- [0098] The microprocessors 236 then acquire a terminal (S310). In specific terms, the mi-

croprocessors 236 acquire a specific job control block which is stored in the microprocessor package 230 as a terminal. 'Terminal' signifies a search processing start value.

[0099] The microprocessors 236 then acquire an element which precedes the terminal acquired in step S310 (S312). More specifically, the microprocessors 236 acquire job control blocks closer to the LRU than the job control blocks acquired in step S310 (S314).

[0100] The microprocessors 236 then determine whether or not the JCB number of the job control blocks acquired in step S312 is a valid JCB number (S314). Numbers in a quantity ranging from 0 to a prescribed number are assigned to the JCB numbers. The microprocessors 236 therefore determine whether or not the JCB number is a valid JCB number depending on whether the JCB number exceeds the prescribed number. In other words, if the JCB number is no more than the prescribed number, the JCB number is validated, and if the JCB number is larger than the prescribed number, the JCB number is invalidated.

[0101] Furthermore, if it is determined in step S314 that the JCB number is valid, the microprocessors 236 then store the JCB numbers and generation numbers of the job control blocks acquired in the search list (S316). In specific terms, the microprocessors 236 add the MP numbers of the microprocessors 236 pertaining to the job control blocks searched for in the search list shown in Fig. 19, as well as the JCB numbers and the generation numbers.

[0102] If it determined in step S314 that a JCB number is invalid, the microprocessors 236 execute the processing of step S320.

[0103] The microprocessors 236 then determine whether the variable 'the number of JCBs' is equal to or more than 8K (S318). In step S318, if the variable 'the number of JCBs' is smaller than 8K, the migration processing of step S308 is repeated. In step S318, if the variable 'the number of JCBs' is equal to or more than 8K, the microprocessors 236 determine whether or not the variable 'number of MPs' is equal to or more than four (S320). In step S320, if the variable 'the number of MPs' is smaller than four, the processing of step S306 and beyond is repeated. In step S320, if the variable 'the number of MPs' is equal to or more than four, the processing is ended.

[0104] (5) Effect of the embodiment

As described hereinabove, with the storage apparatus 10 according to this embodiment, if there is an I/O request from a host 20, the microprocessors 236 store the access range of the I/O request in the job control blocks, turn ON the extent significant bit of the job control blocks, lock the counter value areas of the logical volumes which are access targets of the I/O requests of the MP shared area 1101, acquire the counter values stored in the counter value areas, store the counter values in the job control

blocks, add '1' to (increment by one) the counter values of the MP shared area 1101, and then release the lock on the counter value areas.

[0105] Furthermore, when executing their own I/O requests, the microprocessors 236 search for the I/O queues (LDEV queues) of the access-target logical volumes, compare the counter values of the searched for I/O requests with the counter values of their own I/O requests, and if the counter values of their own I/O requests are greater than the counter values of the searched for I/O requests, determine whether or not there is an extent range overlap, executing I/O processing if there is no extent range overlap and waiting to execute I/O processing if there is an extent range overlap.

[0106] Thus, according to this embodiment, when executing I/O processing, the microprocessors 236 are able to provide a counter value to each of the logical volumes which are targets for the I/O processing and determine the access sequence for access to the logical volumes on the basis of these counter values.

[0107] Conventionally access of the other I/O processing to the logical volumes must be restricted while a determination is made as to whether or not there is an extent range overlap, but by providing counter values, the counter values may be acquired and access by other I/O processing may be restricted only while the counter values are being incremented. It is therefore possible to execute, in parallel, exclusive extent processing of a plurality of I/O requests with respect to the same logical volume, thereby improving the response time performance.

[0108] (6) Other Embodiments

Note that although a case was described in the above embodiment in which the microprocessors 236 implement exclusive extent processing and I/O processing and so forth on the basis of various programs stored in memory such as the logical memory 231 and flash memory 234 in the microprocessor packages 230, the present invention is not limited to such a case, for example various functions may be implemented by storing various programs in storage devices outside the microprocessor package 230 and then calling up these programs.

[0109] Moreover, in the above embodiment, a case was described in which four microprocessors 236 are provided in the microprocessor packages 230, but the present invention is not limited to this case, for example more than four microprocessors 236 may be provided in the microprocessor packages 230 and the exclusive extent processing according to this embodiment may be executed in each of the microprocessors 236.

[0110] Moreover, in the above embodiment, a case was described where the ownership rights of the microprocessor packages 230 are configured for each of the logical volumes, but the present invention is not limited to such a case; for example, if a virtual volume is provided to the hosts 20 by means of a Thin Provisioning function,

ownership rights to the microprocessor packages 230 may also be configured for each virtual volume.

Industrial Applicability

- [0111] The present invention can be applied to a storage apparatus with which the response time performance can be improved by executing, in parallel, exclusive extent processing of a plurality of I/O requests.

Reference Signs List

- [0112] 1 Computer system
10 Storage apparatus
15 Storage medium
20 Host
30 Management server
40, 50 Network
100, 101 Module
110 Control module
120 Driver module
210 Channel adapter
220 Disk adapter
230 Microprocessor package
231 Logical memory
232 Memory controller hub
233 I/O controller hub
234 Flash memory
235 CPU
236 Microprocessor package
237 Front side bus
240 Cache package

Claims

- [Claim 1] A storage apparatus which is connected via a network to a host apparatus which requests data I/Os, comprising:
a plurality of microprocessors which control data I/O to and from a plurality of storage devices on the basis of the data I/O requests from the host apparatus;
a first memory which stores, for each storage area of a predetermined range provided by the plurality of storage devices, counter values indicating the order of the I/Os to and from the storage areas; and
a second memory which stores control information of the I/O requests for each of the I/O requests,
wherein the microprocessors:
store, if there is one I/O request from the host apparatus, an I/O request range of the one I/O request in an area for storing control information of the one I/O request of the second memory,
acquire, from the first memory, the counter value of the storage area which is the target of the one I/O request and stores the counter value in the area for storing control information of the one I/O request of the second memory, and
compare, if the counter value of the one I/O request is greater than the counter value of another I/O request, the I/O request range of the one I/O request stored in the second memory with the I/O request range of the other I/O request.
- [Claim 2] The storage apparatus according to claim 1, wherein the control information of the I/O request stored in the second memory includes information indicating a counter value and an extent range, information indicating whether or not the counter value and extent range are valid values, and information indicating whether or not I/O processing can be executed.
- [Claim 3] The storage apparatus according to claim 2, wherein the microprocessor compares the I/O request range of the one I/O request with the other I/O request range if the counter value and the extent range contained in the control information of the other I/O request are valid.
- [Claim 4] The storage apparatus according to claim 1, wherein the microprocessor acquires the counter value from the first memory and, while the counter value is stored in the second memory, restricts writing to the area for storing the counter value of the storage area which is the

target of the I/O request in the first memory.

[Claim 5] The storage apparatus according to claim 1, wherein the microprocessor acquires the counter value of the storage area which is the target of the one I/O request from the first memory and then, after storing the counter value in the area for storing control information of the one I/O request of the second memory, adds '1' to the counter value acquired from the first memory and stores this value in the first memory.

[Claim 6] The storage apparatus according to claim 1, wherein if, as a result of comparing the I/O request range of the one I/O request stored in the second memory with the I/O request range of the other I/O request, there is no overlap between the I/O request ranges, the microprocessor executes the one I/O request processing, and if there is an overlap between the I/O request ranges, the microprocessor places the processing of the one I/O request on standby.

[Claim 7] The storage apparatus according to claim 6, wherein, if the microprocessor executes the one I/O request processing, the microprocessor stores information indicating that I/O processing can be executed in the control information of the one I/O request, and if the microprocessor places the processing of the one I/O request on standby, the microprocessor stores information indicating that the microprocessor has placed the execution of the I/O processing on standby in the control information of the one I/O request.

[Claim 8] The storage apparatus according to claim 1, wherein, if the I/O queue of the I/O request linked to each of the storage areas is searched starting with the MRU and moving toward the LRU and the counter value of a specific I/O request is greater than the counter value of an I/O request corresponding to the searched for I/O queue, the microprocessor compares the I/O request range of the specific I/O request stored in the second memory with an I/O request range of the searched for I/O request.

[Claim 9] The storage apparatus according to claim 8, wherein the microprocessor creates a search list which indicates the search order for searching for I/O queues in the storage areas which are the access targets of the I/O request before searching, from the MRU toward the LRU, for the I/O queues of each of the storage areas and comparing the I/O request range of the I/O request.

[Claim 10] The storage apparatus according to claim 9, wherein the micro-

[Claim 11]

processor adds identification information, which identifies a plurality of I/O queues of each of the storage areas, to the search list.

A data access method which employs a storage apparatus which is connected via a network to a host apparatus that requests data I/Os, the storage apparatus including:

a plurality of microprocessors which control data I/O to and from a plurality of storage devices on the basis of the data I/O requests from the host apparatus;

a first memory which stores, for each storage area of a predetermined range provided by the plurality of storage devices, counter values indicating the order of the I/Os to and from the storage areas; and

a second memory which stores control information of the I/O requests for each of the I/O requests,

the data access method comprising steps in which the microprocessors: store, if there is one I/O request from the host apparatus, an I/O request range of the one I/O request in an area for storing control information of the one I/O request of the second memory;

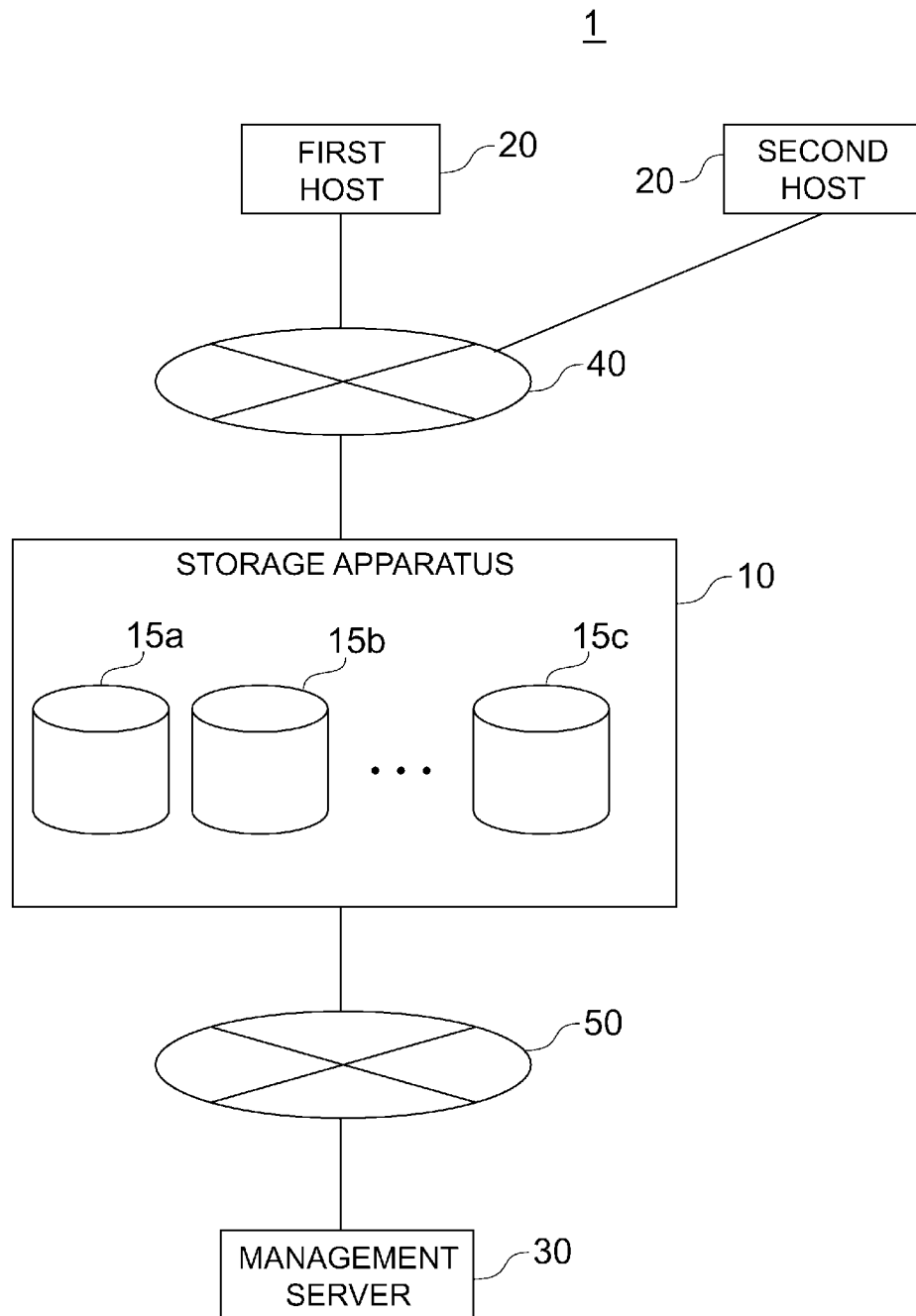
acquire, from the first memory, the counter value of the storage area which is the target of the one I/O request;

store the counter value in the area for storing control information of the one I/O request of the second memory; and

compare, if the counter value of the one I/O request is greater than the counter value of another I/O request, the I/O request range of the one I/O request stored in the second memory with the I/O request range of the other I/O request; and

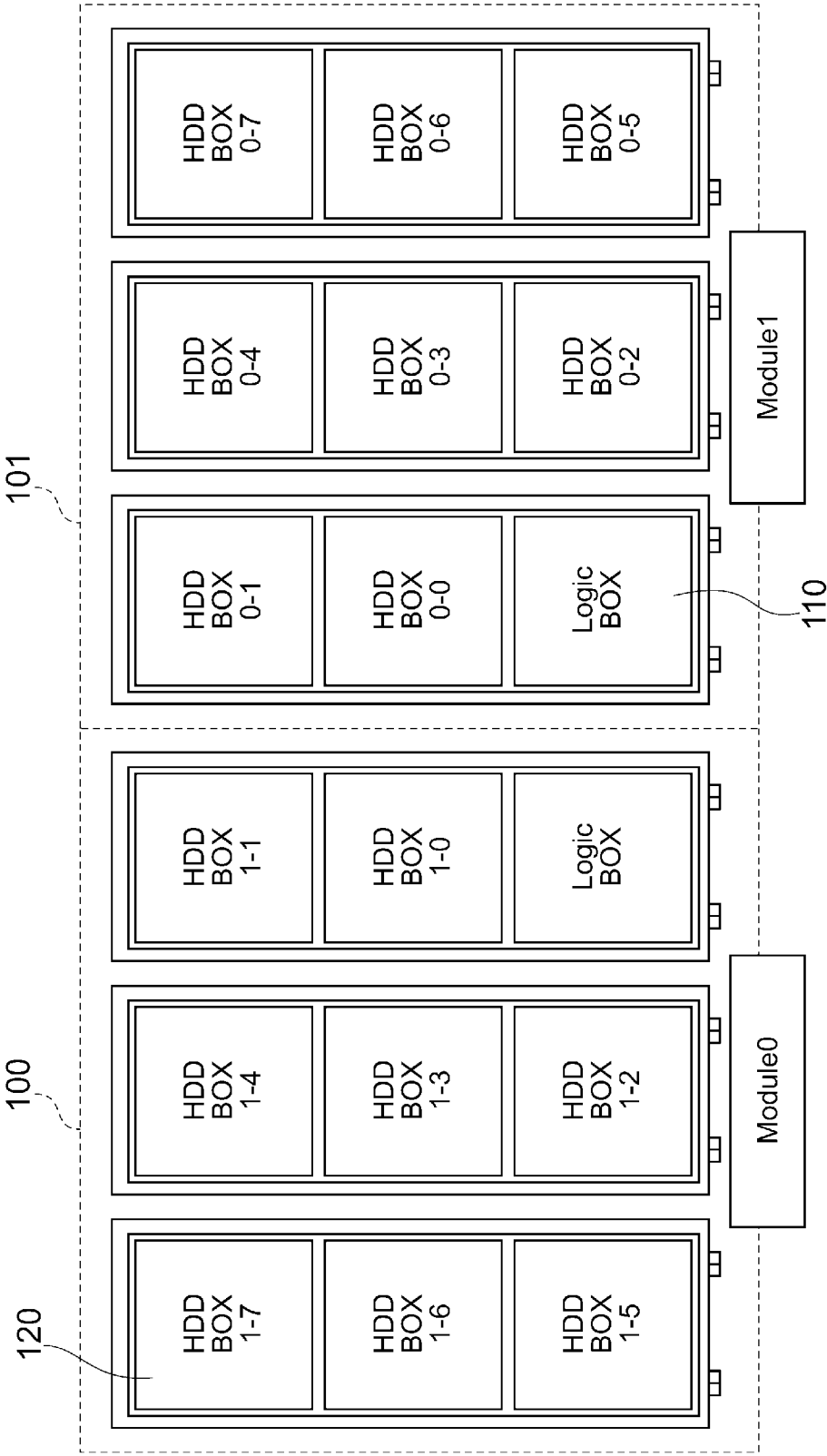
if, as a result of comparing the I/O request ranges, there is no overlap between the I/O request ranges, the microprocessors execute the processing of the one I/O request, and if there is an overlap between the I/O request ranges, the microprocessors place the processing of the one I/O request on standby.

[Fig. 1]

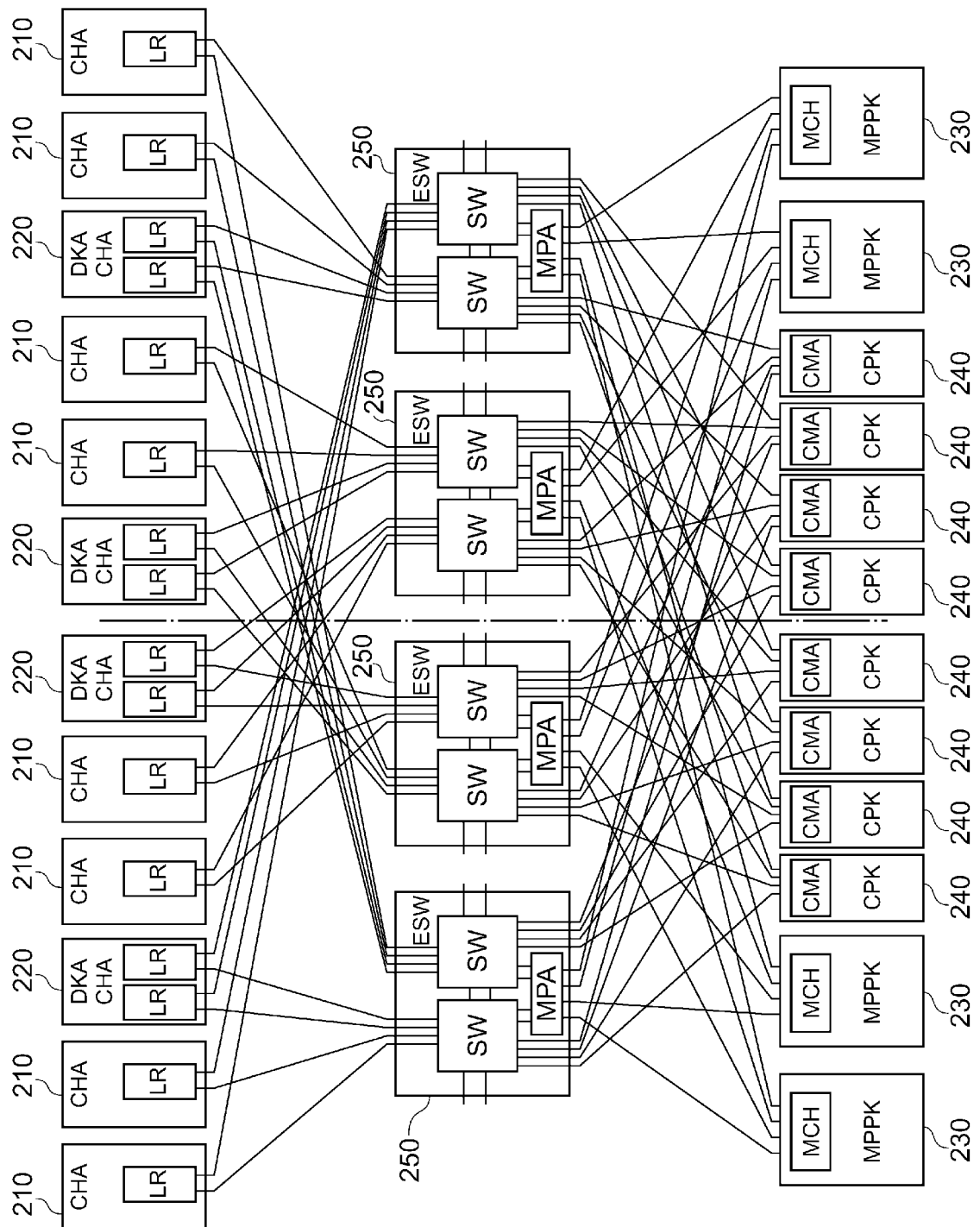
FIG. 1

[Fig. 2]

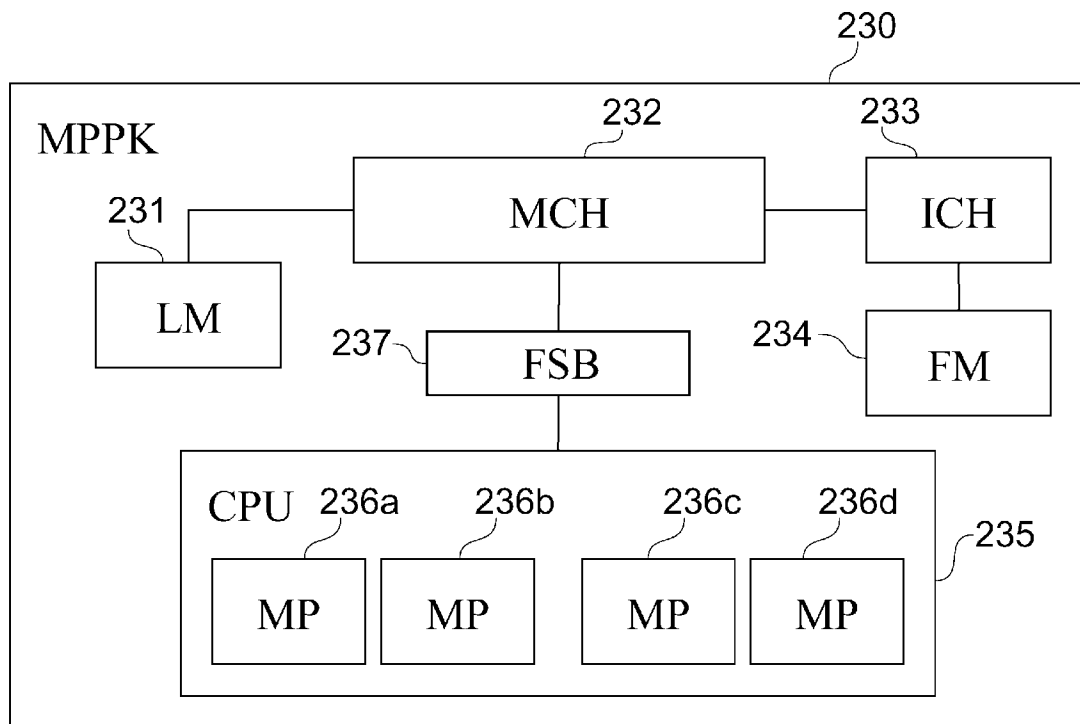
FIG. 2



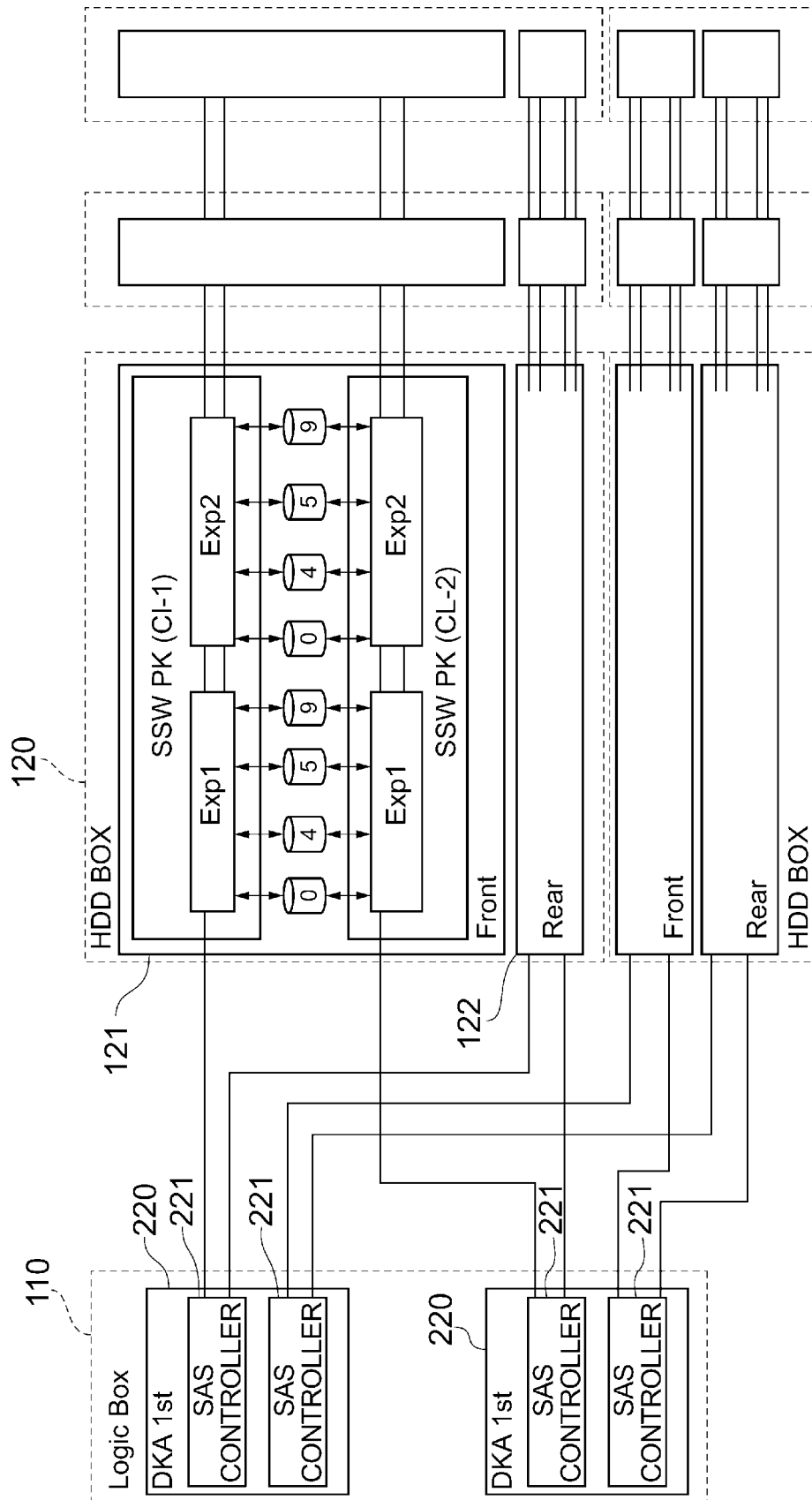
[Fig. 3]

FIG. 3

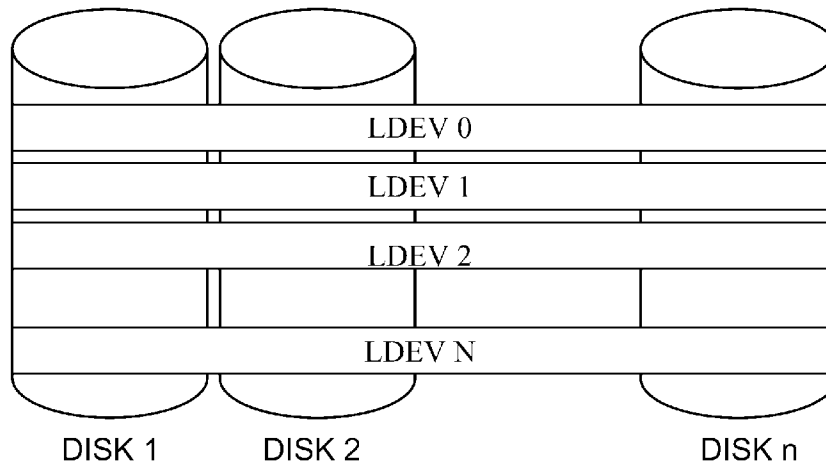
[Fig. 4]

FIG. 4

[Fig. 5]

FIG. 5

[Fig. 6]

FIG. 6

[Fig. 7]

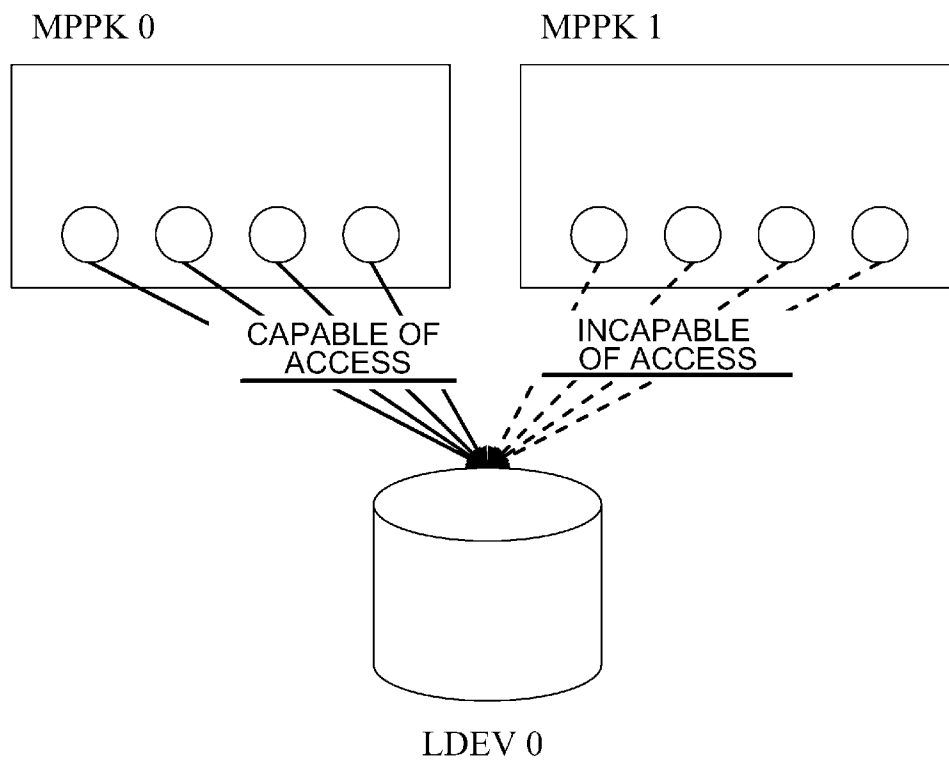
FIG. 7

LDEV No	No. OF MPPK WITH OWNERSHIP RIGHTS
0	0
1	1
2	0
⋮	⋮

700

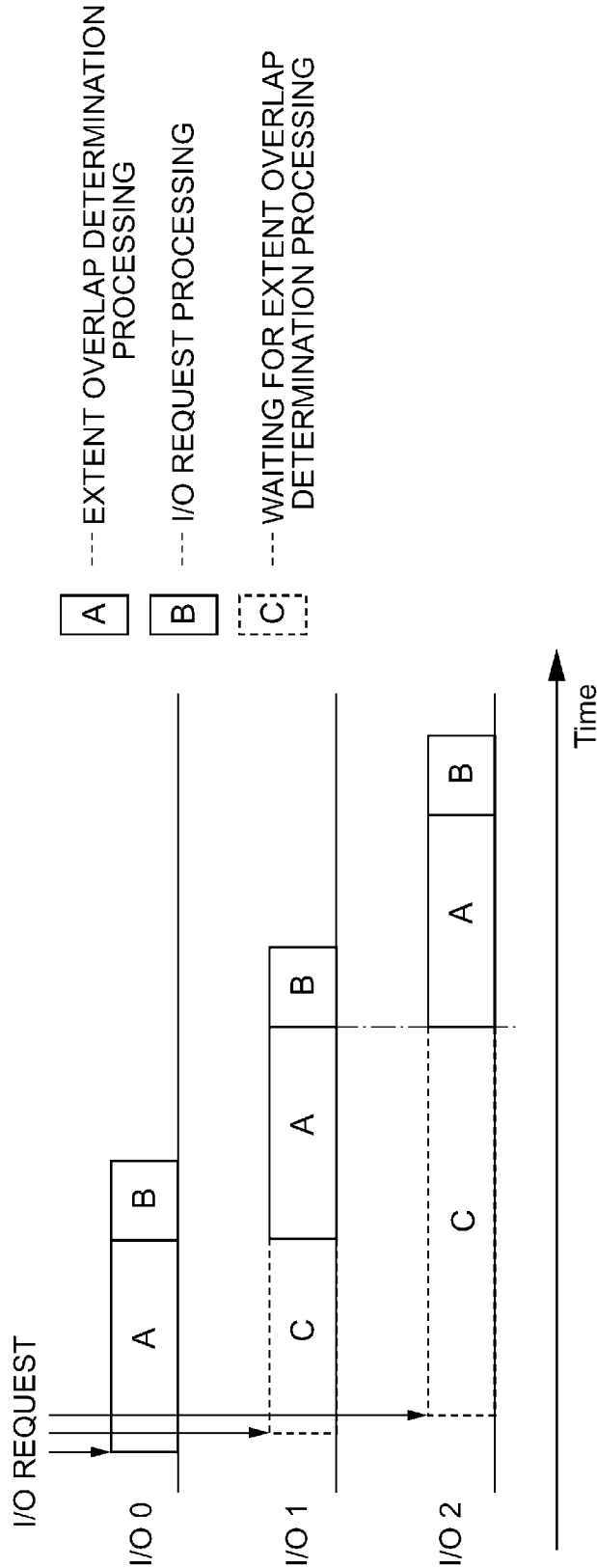
701 702

[Fig. 8]

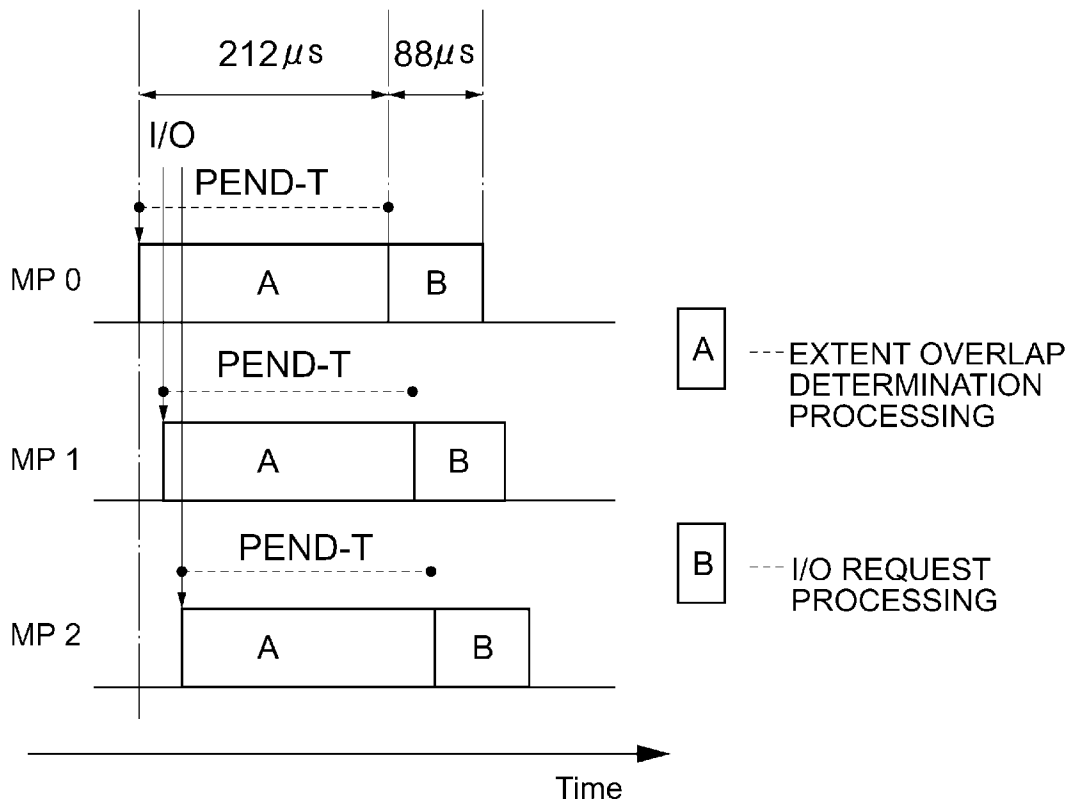
FIG. 8

[Fig. 9]

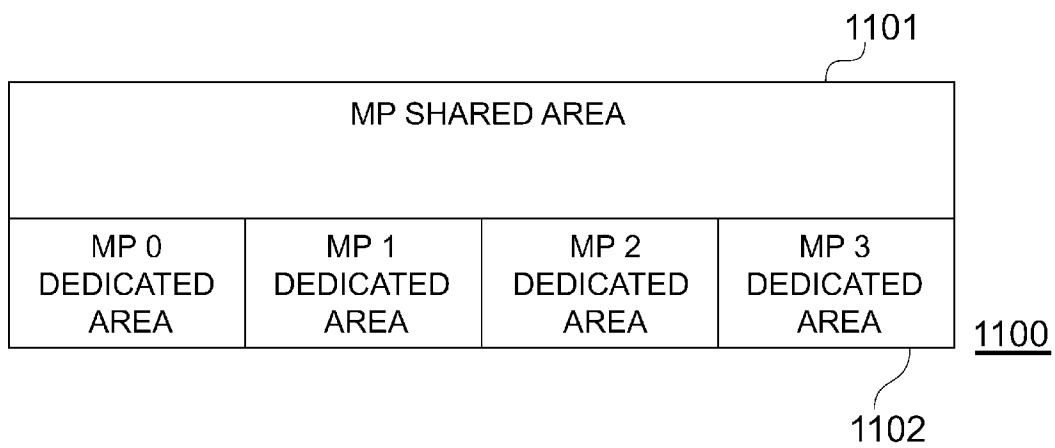
FIG. 9



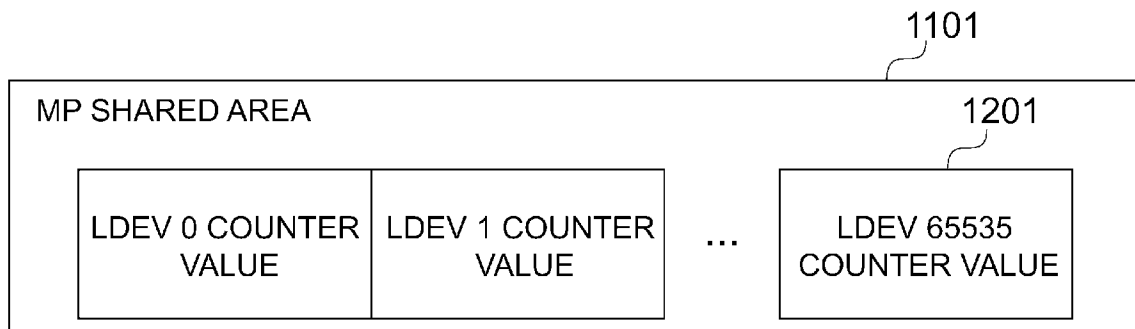
[Fig. 10]

FIG. 10

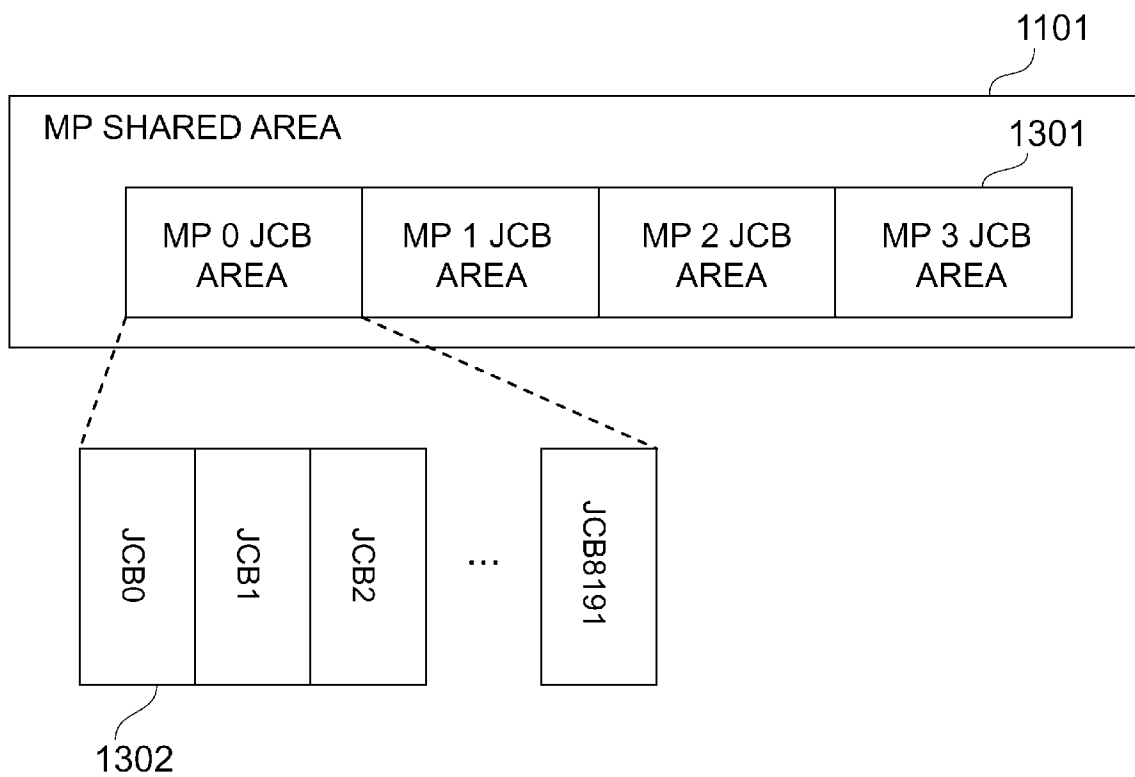
[Fig. 11]

FIG. 11

[Fig. 12]

FIG. 12

[Fig. 13]

FIG. 13

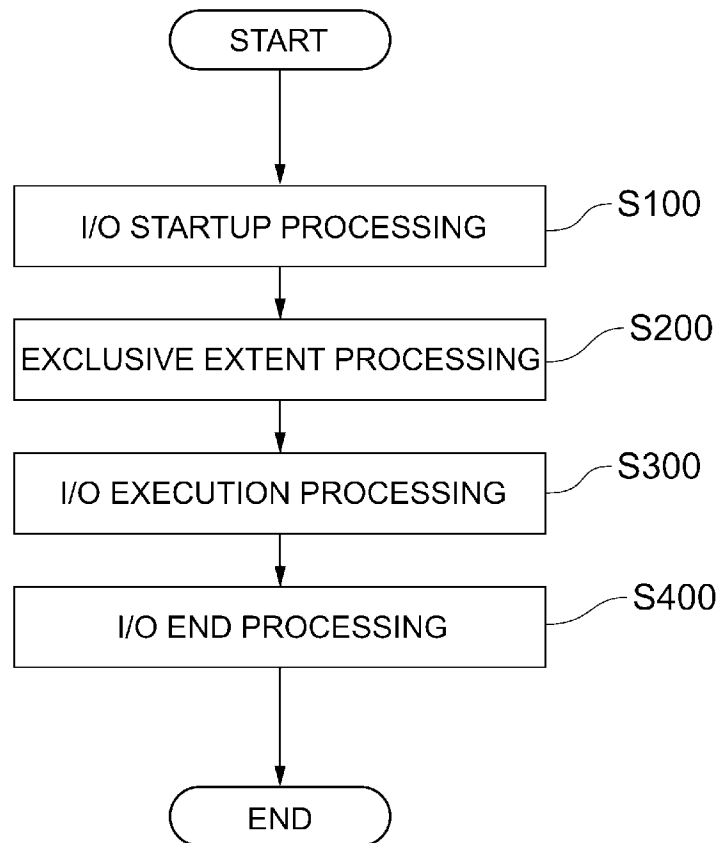
[Fig. 14]

FIG. 14

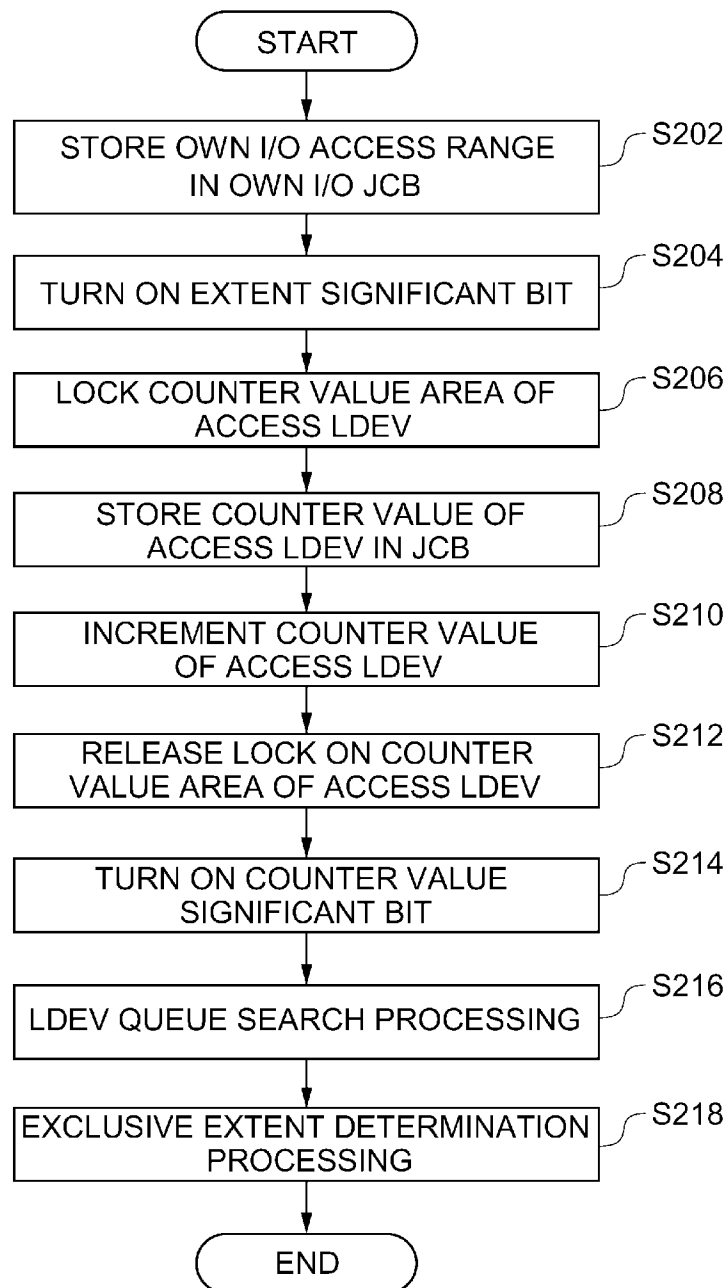
No	ITEM	REMARKS
1	EXTENT PROCESSING CONTROLLER	EXTENT RESERVATION BIT 0x80000000 EXCLUSIVE EXTENT WAITING BIT 0x40000000 EXTENT SIGNIFICANT BIT 0x20000000 COUNTER VALUE SIGNIFICANT BIT 0x10000000
2	COUNTER VALUE	
3	START POSITION ADDRESS	
4	END POSITION ADDRESS	
1401	1402	1403

1302

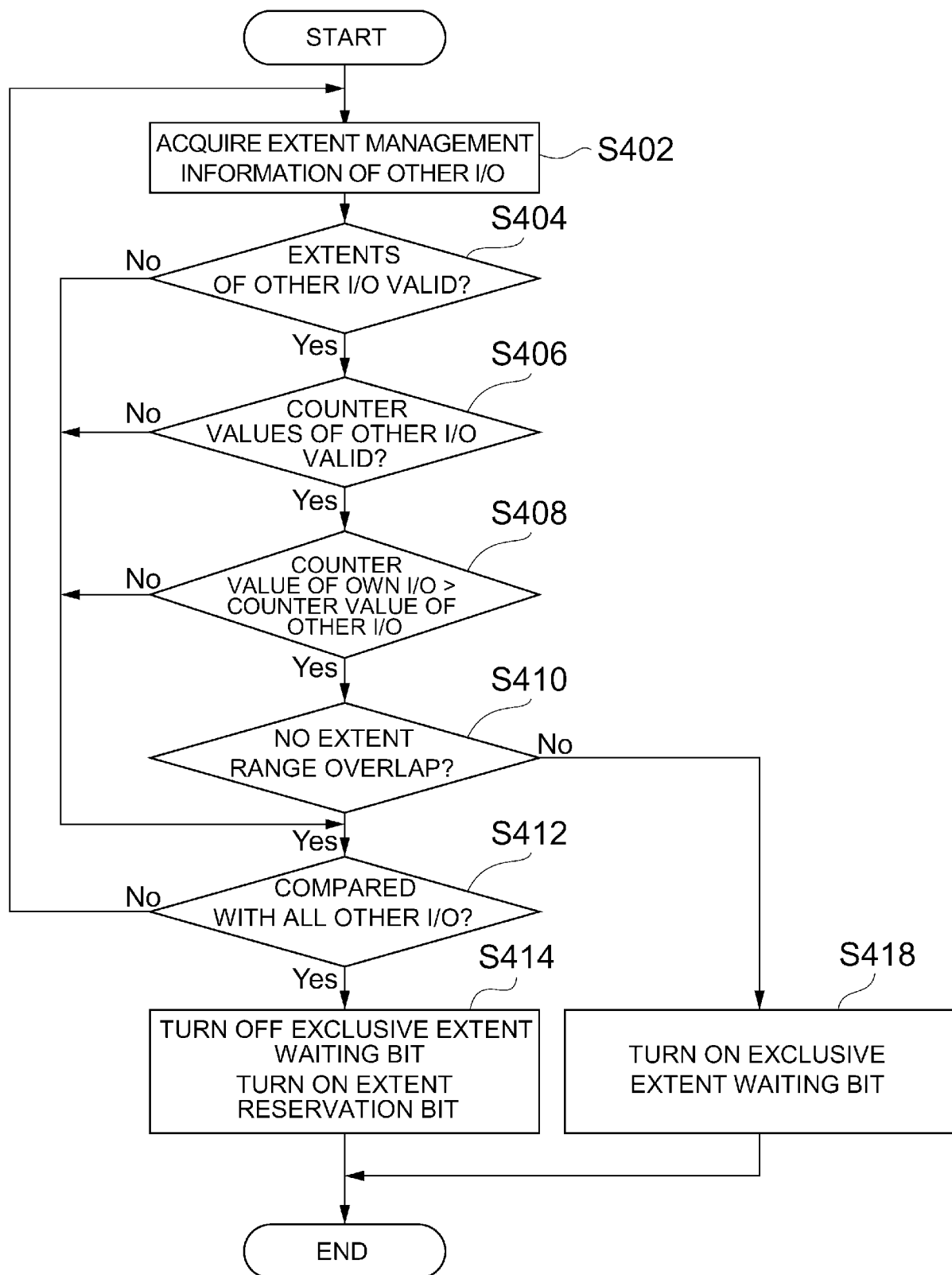
[Fig. 15]

FIG. 15

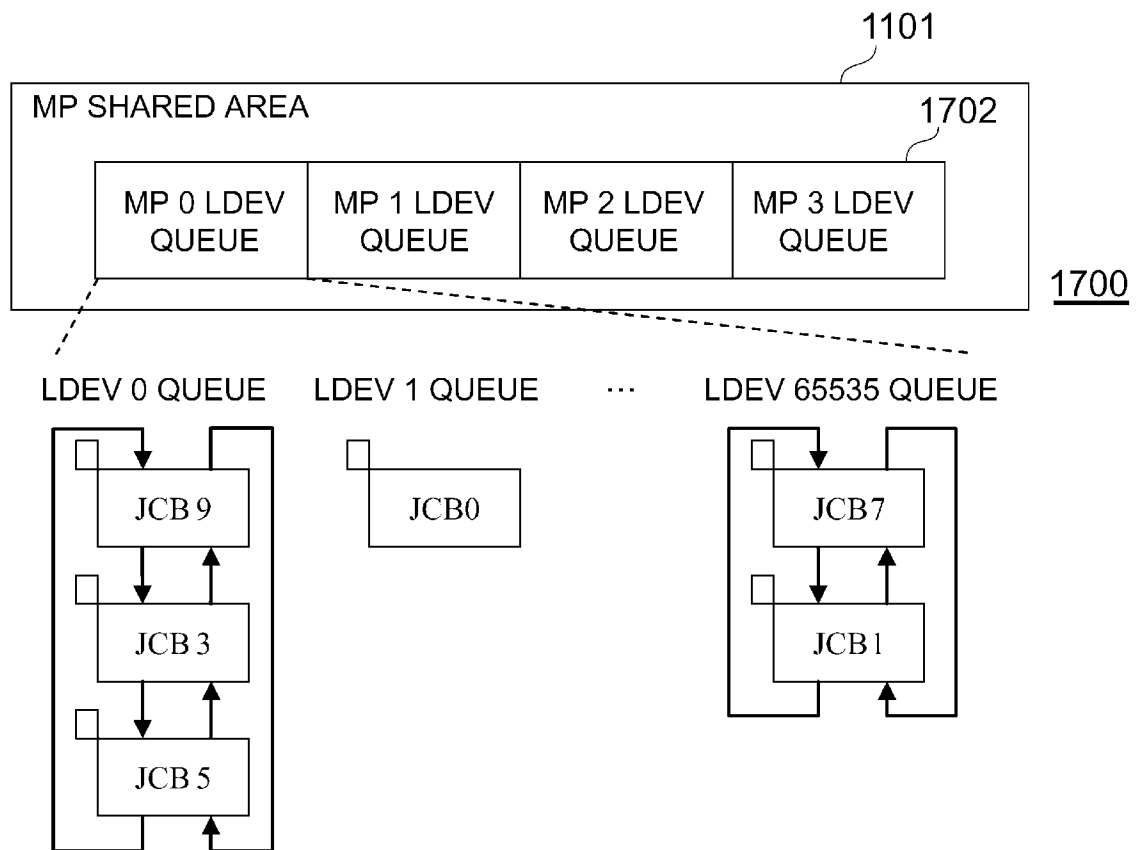
[Fig. 16]

FIG. 16

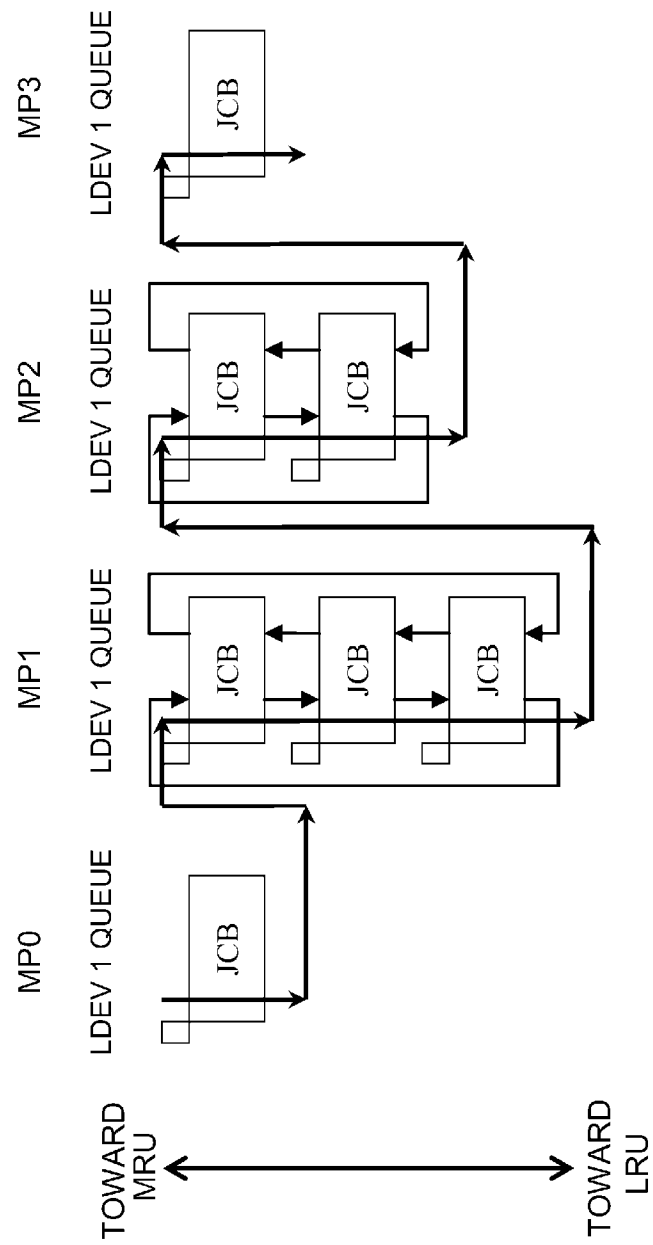
[Fig. 17]

FIG. 17

[Fig. 18]

FIG. 18

[Fig. 19]

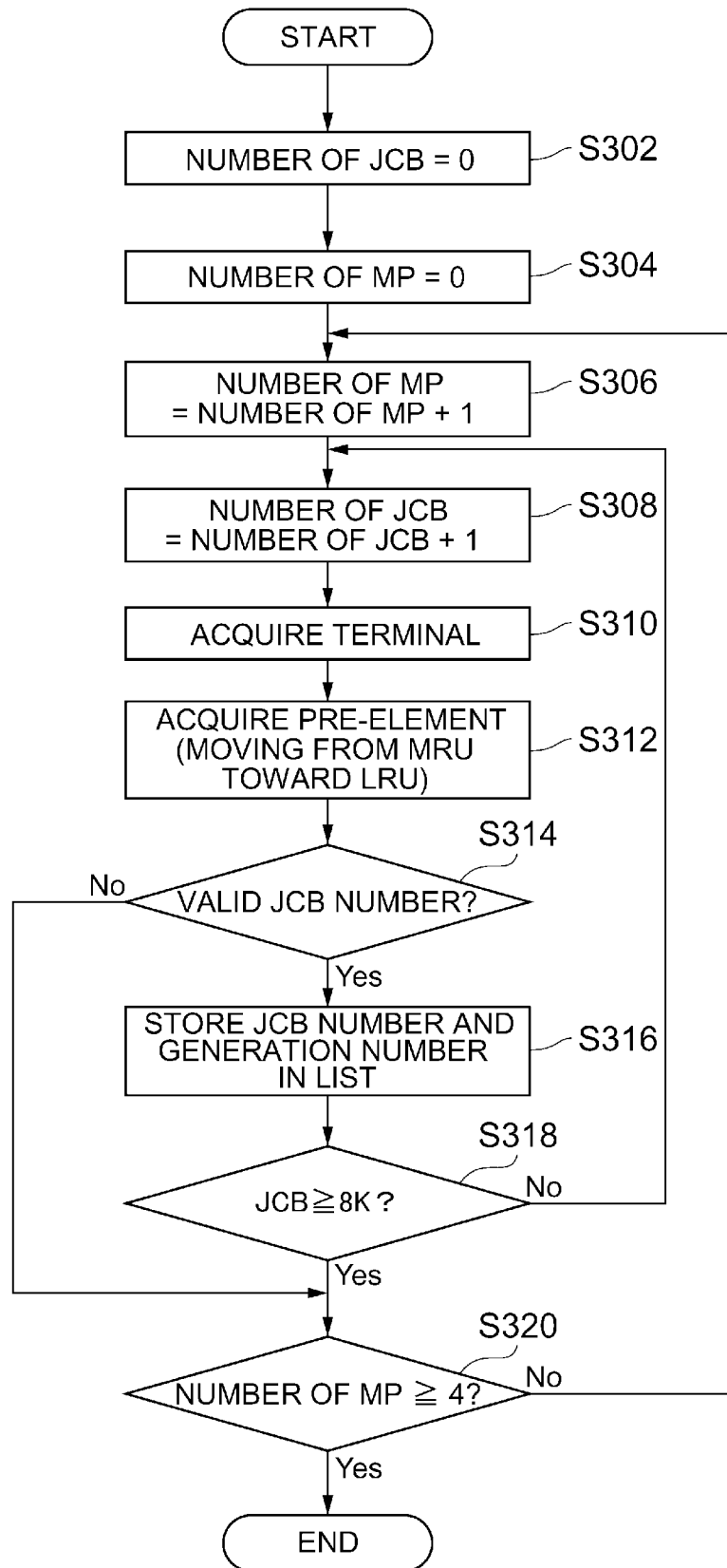
FIG. 19

[Fig. 20]

FIG. 20

MP No	JCB No	GENERATION No	<u>1900</u>
0	9	0	
0	3	1	
0	5	1	
1	0	0	
⋮	⋮	⋮	
1901	1902	1903	

[Fig. 21]

FIG. 21

INTERNATIONAL SEARCH REPORT

International application No
PCT/JP2010/005403

A. CLASSIFICATION OF SUBJECT MATTER
INV. G06F3/06
ADD.

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)
G06F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practical, search terms used)

EPO-Internal, WPI Data

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2003/204691 A1 (CHANG JOON [US] ET AL) 30 October 2003 (2003-10-30) figures 1,2 paragraph [0022] - paragraph [0036] -----	1-11
A	EP 1 986 090 A2 (HITACHI LTD [JP]) 29 October 2008 (2008-10-29) the whole document -----	1-11



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 document 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

11 April 2011

Date of mailing of the international search report

18/04/2011

Name and mailing address of the ISA/

European Patent Office, P.B. 5818 Patentlaan 2
NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Authorized officer

Andlauer, J

INTERNATIONAL SEARCH REPORT

Information on patent family members

International application No

PCT/JP2010/005403

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2003204691	A1	30-10-2003	NONE

EP 1986090	A2	29-10-2008	CN 101295227 A 29-10-2008
			JP 2008269424 A 06-11-2008
			US 2008263190 A1 23-10-2008
