US 20040210714A1

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2004/0210714 A1

Kochiya (43) Pub. Date: Oct. 21, 2004

(54) RAID APPARATUS STORING A PLURALITY OF SAME LOGICAL VOLUMES ON DIFFERENT DISK UNITS

(75) Inventor: **Toshiaki Kochiya**, Kawasaki-shi (JP)

Correspondence Address:
STAAS & HALSEY LLP
SUITE 700
1201 NEW YORK AVENUE, N.W.
WASHINGTON, DC 20005 (US)

(73) Assignee: **FUJITSU LIMITED**, Kawasaki (JP)

(21) Appl. No.: **10/843,309**

(22) Filed: **May 12, 2004**

Related U.S. Application Data

(62) Division of application No. 10/259,422, filed on Sep. 30, 2002, which is a division of application No. 08/904,346, filed on Jul. 31, 1997.

(30) **Foreign Application Priority Data**

Nov. 1, 1996 (JP) 8-291692

Publication Classification

(51) **Int. Cl.**[7] **G06F 12/00**

(52) **U.S. Cl.** **711/114**; 711/170

(57) ABSTRACT

A RAID apparatus includes a plurality of same logical volumes allocated on a real volume. The real volume is so designed that a plurality of same logical volumes are respectively allocated on different physical disk units and a combination of a plurality of logical volumes allocated on each physical disk unit differs from one physical disk unit to another. This structure prevents uneven loading on the real volume from occurring due to uneven loads on the logical volumes.

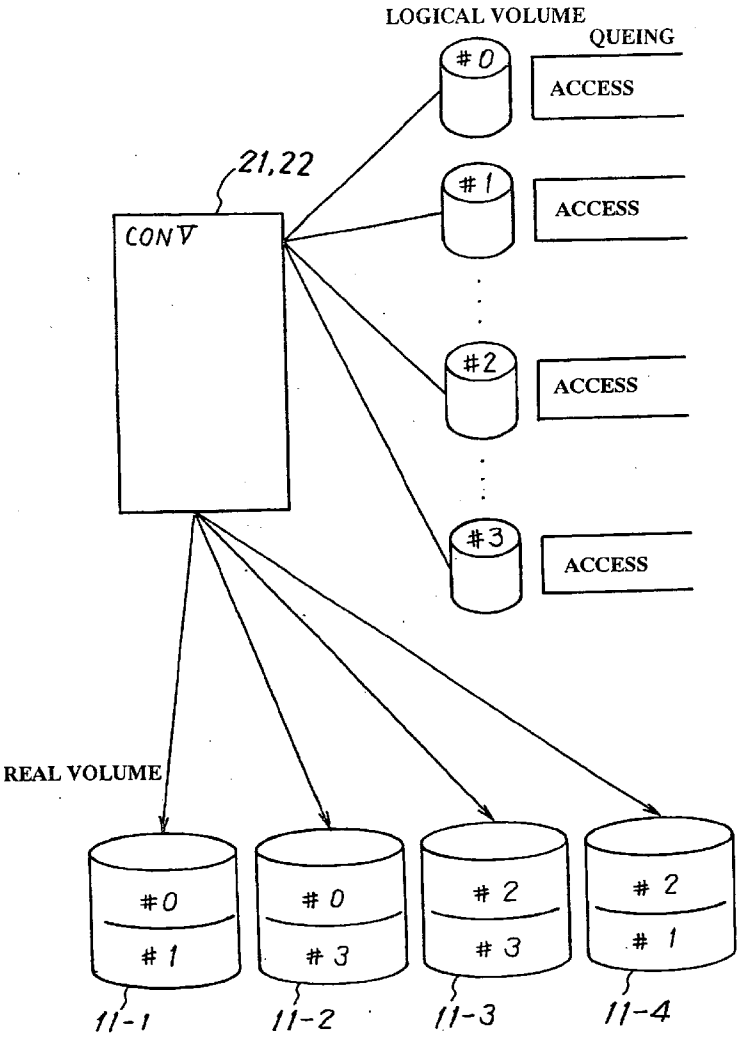


FIG. 1

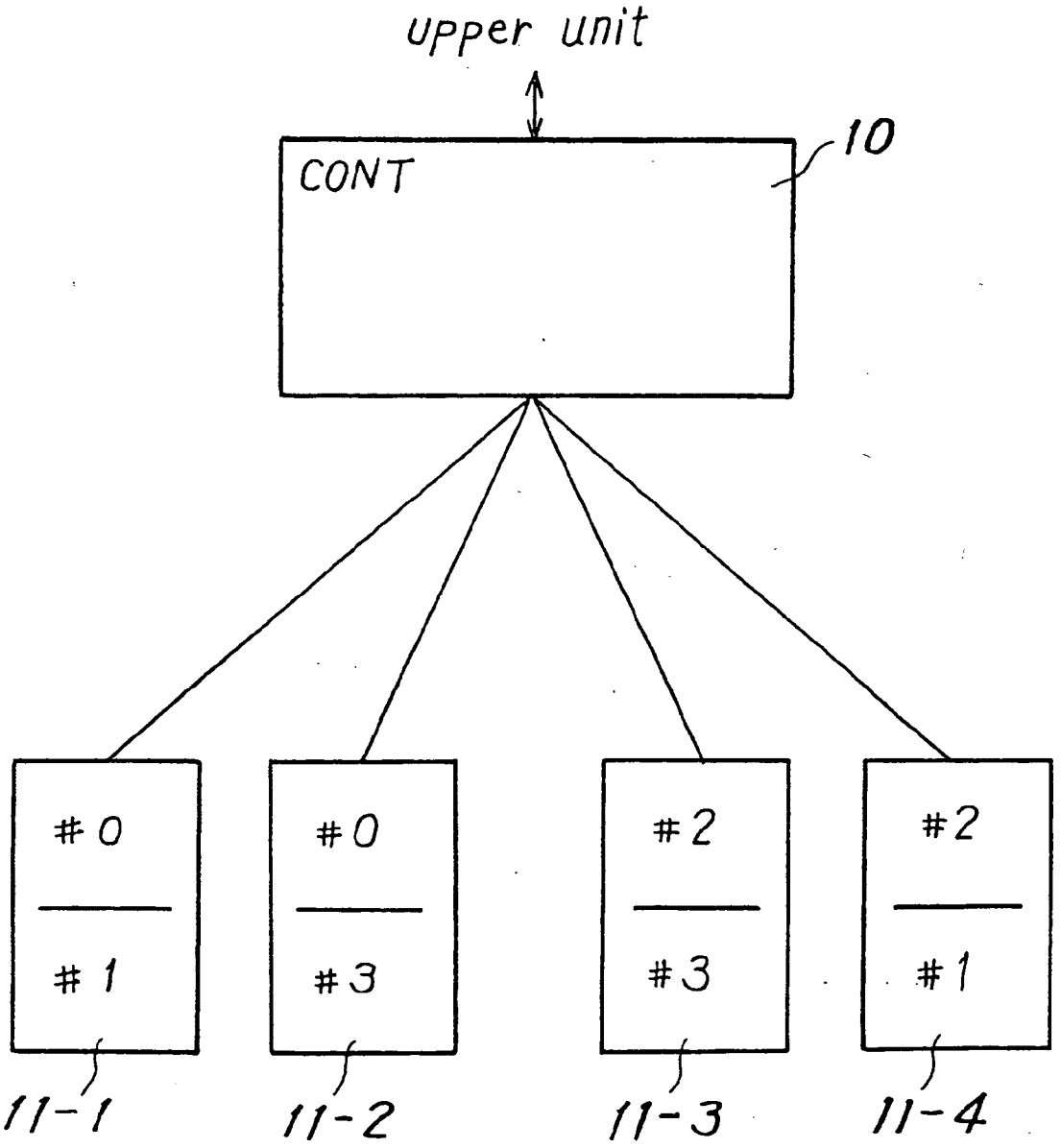


FIG. 2

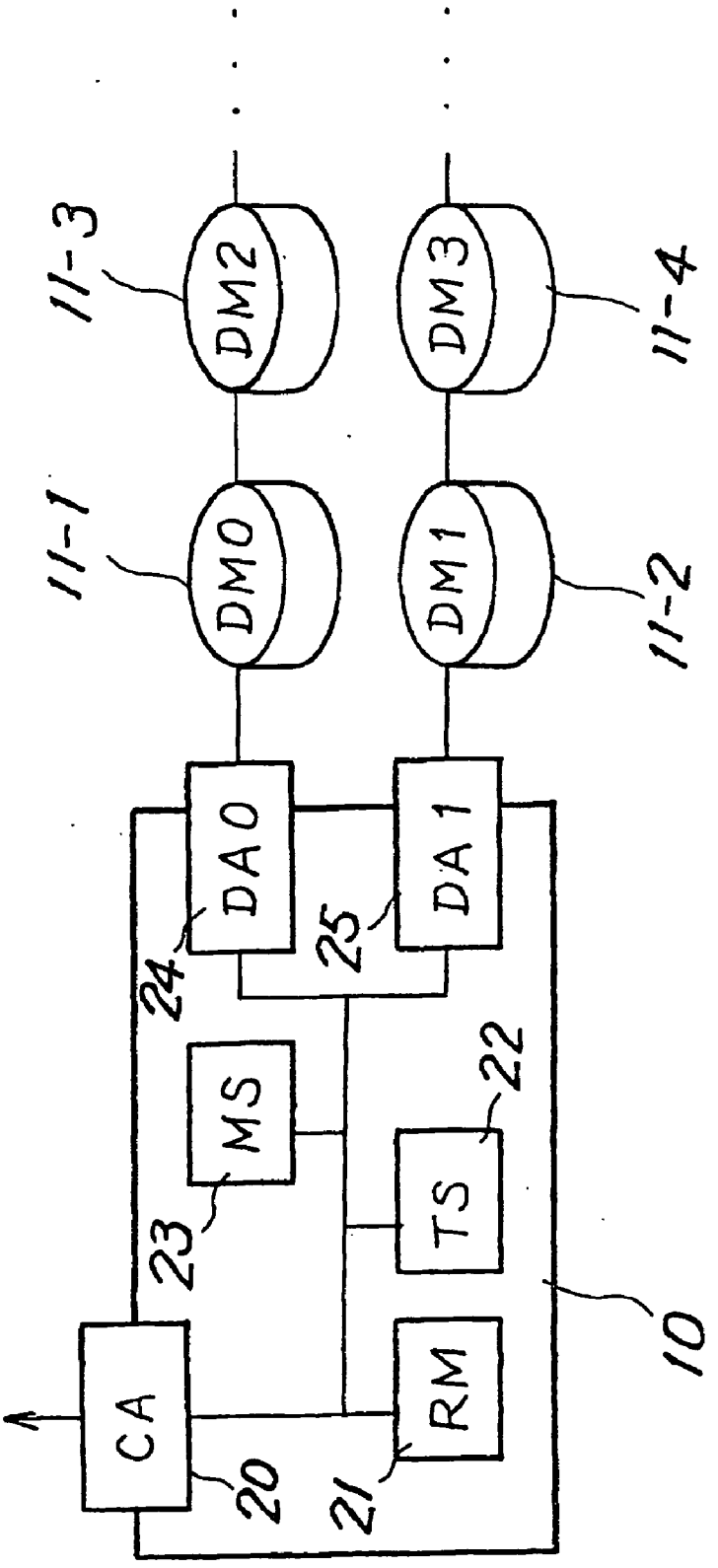

11-3
DM2
DM3
11-4
11-1
DM0
DM1
11-2
DA 0
DA 1
24
25
MS
22
23
TS
10
CA
RM
20
21

FIG. 3

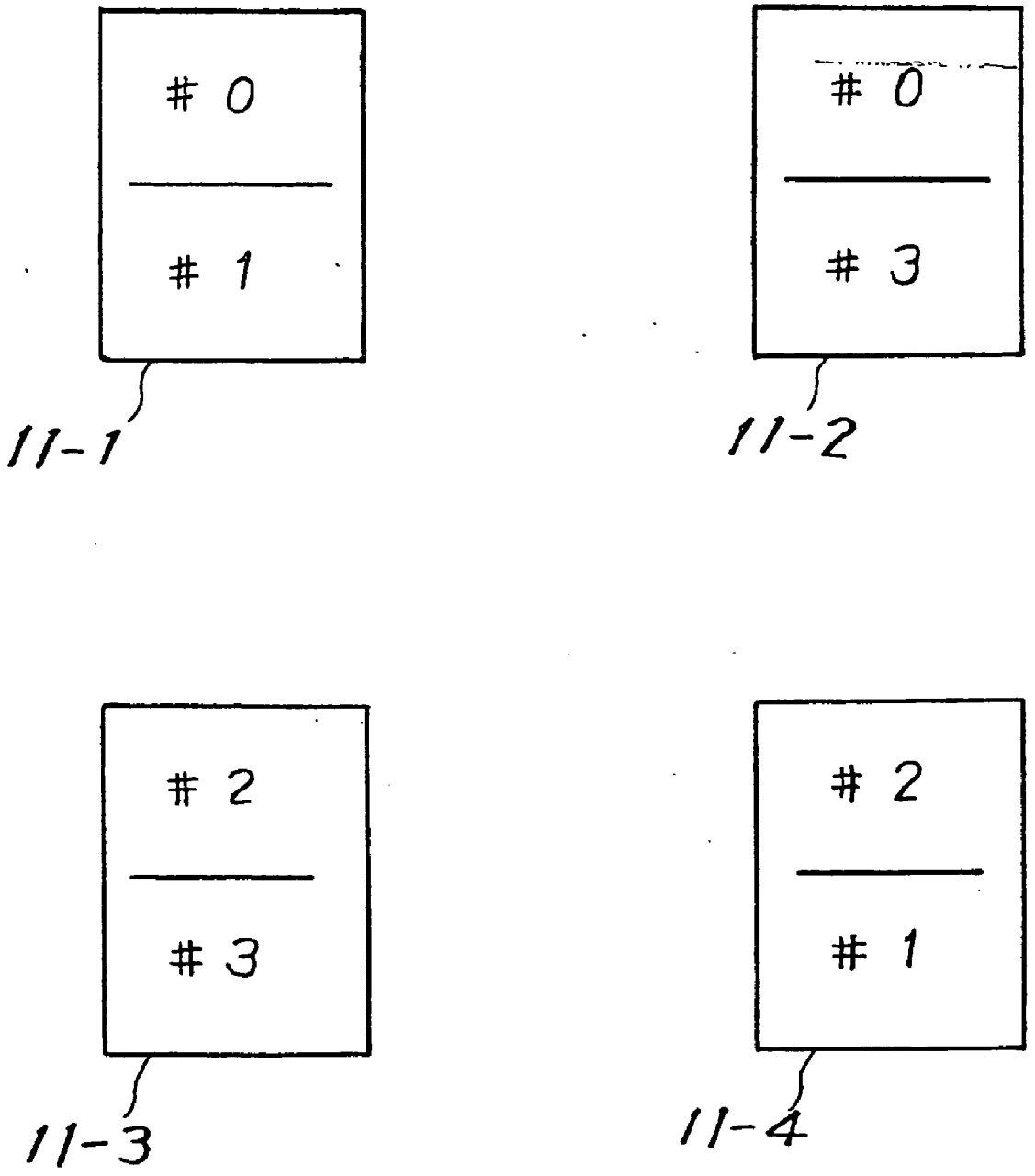


FIG. 4

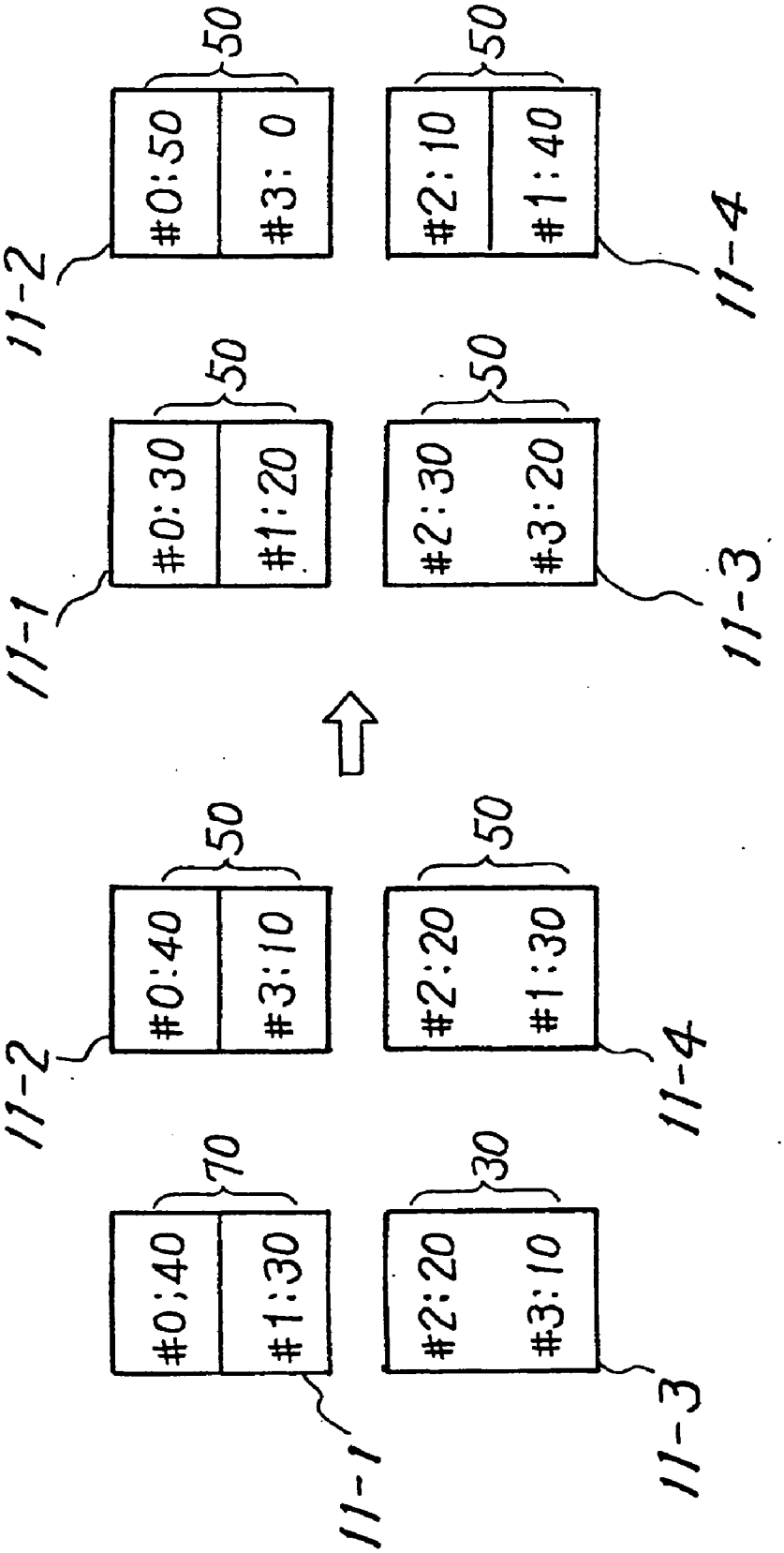

11-2
#0:50
#3: 0
50
11-4
#2:10
#1:40
50
11-1
#0:30
#1:20
50
11-3
#2:30
#3:20
50
11-2
#0:40
#3:10
50
11-4
#2:20
#1:30
50
11-1
#0:40
#1:30
70
11-3
#2:20
#3:10
30

FIG. 5

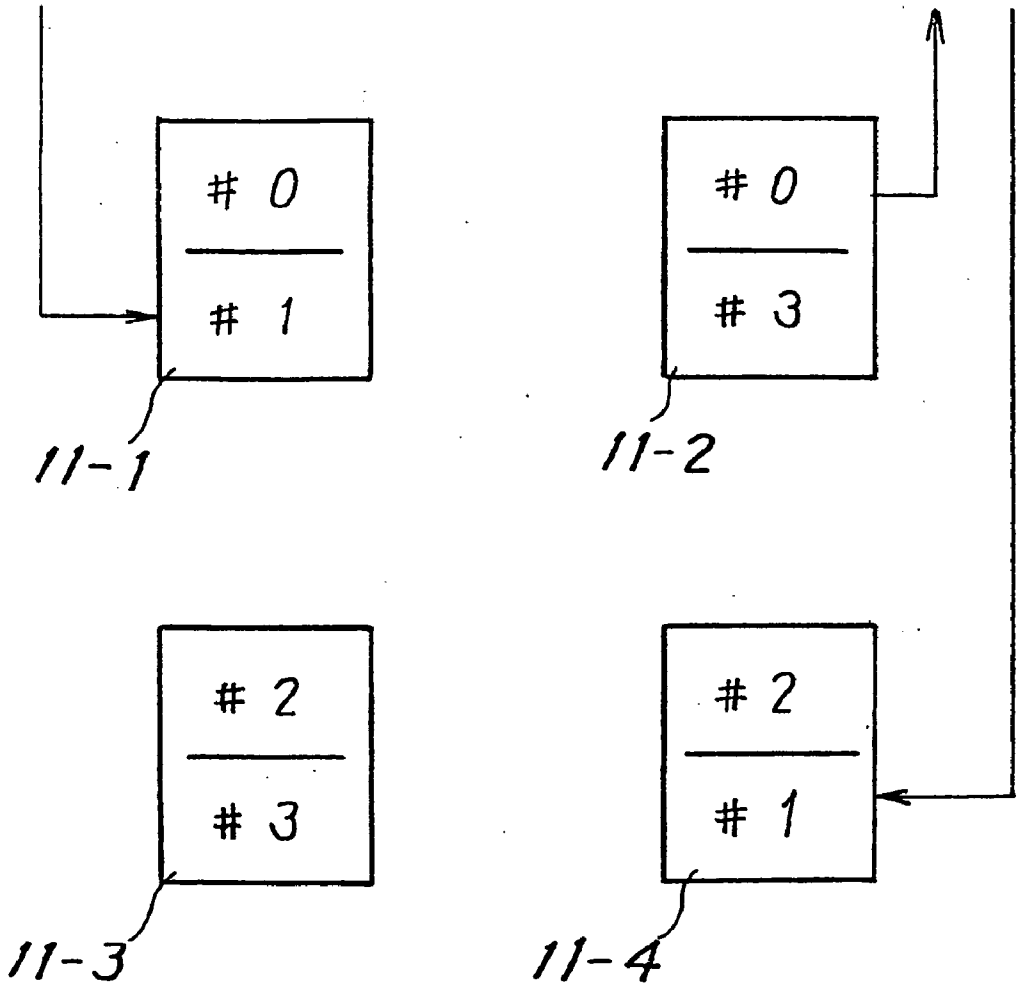


FIG. 6

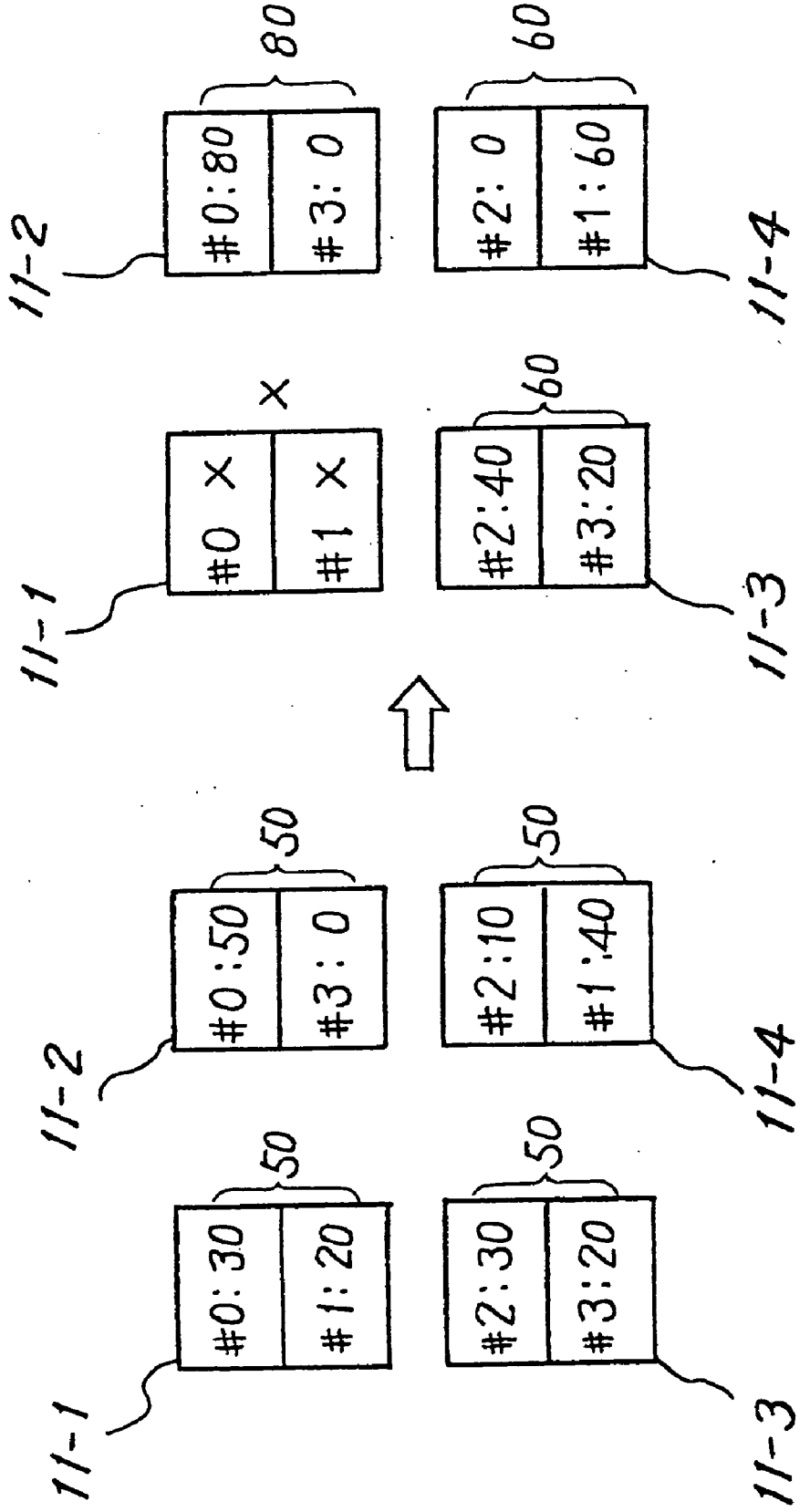

11-2
#0:80
#3: 0
80
11-4
#2: 0
#1:60
60
11-1
#0 X
#1 X
X
11-3
#2:40
#3:20
60
11-2
#0:50
#3: 0
50
11-4
#2:10
#1:40
50
11-1
#0: 30
#1: 20
50
11-3
#2: 30
#3:20
50

FIG. 7

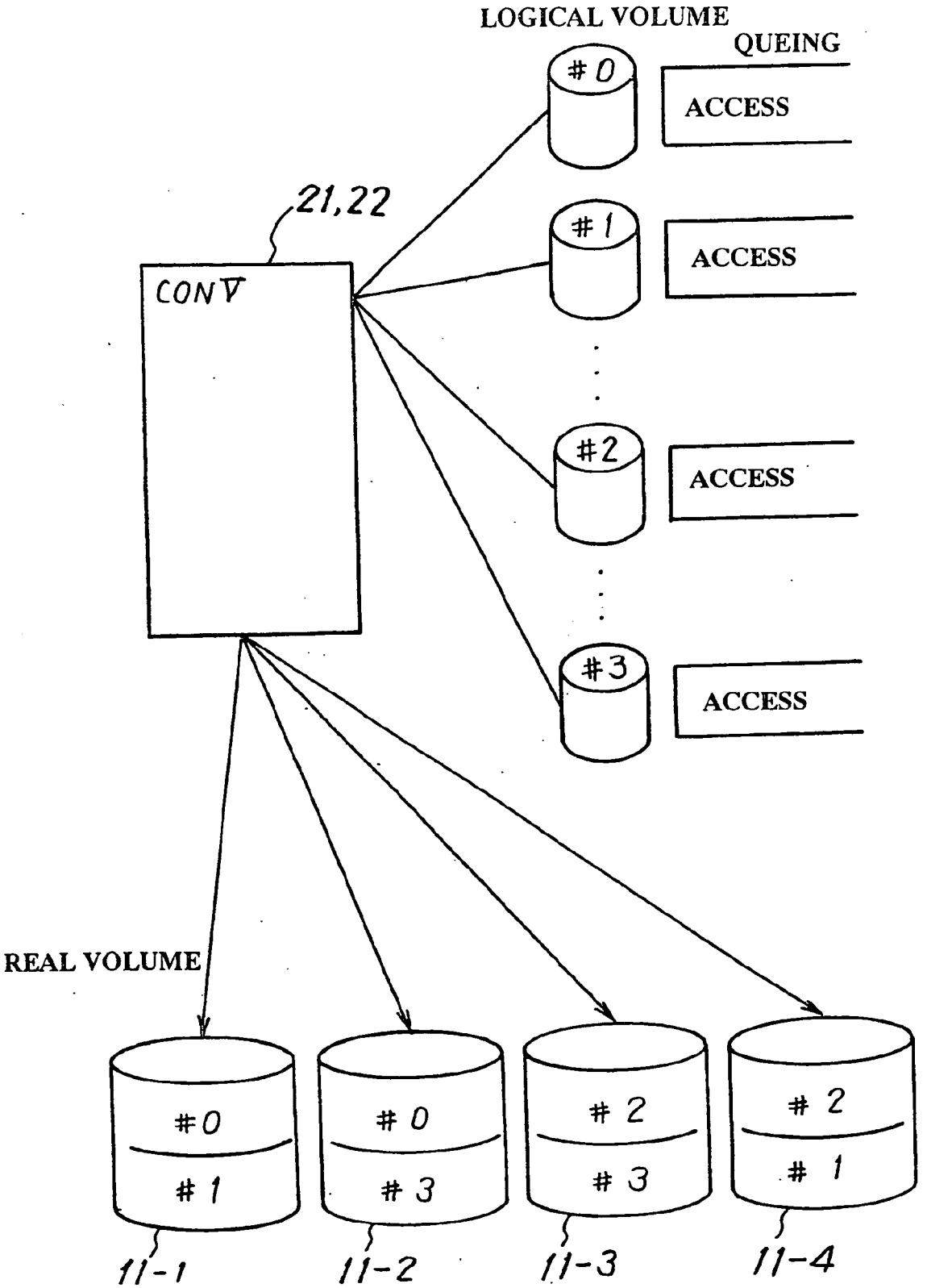

LOGICAL VOLUME
QUEING
#0
ACCESS
21,22
#1
ACCESS
CONT
#2
ACCESS
#3
ACCESS
REAL VOLUME
#0
#1
#0
#3
#2
#3
#2
#1
11-1
11-2
11-3
11-4

FIG. 8A

Logical Volume #0	Real Volume Number	----------	Real Volume Number	Counter
⋮	⋮	⋮	⋮	⋮
Logical Volume #3	Real Volume Number	----------	Real Volume Number	Counter

22-1

FIG. 8B

Real Volume 1	In Use Flag
⋮	⋮
Real Volume 4	In Use Flag

22-2

FIG. 9

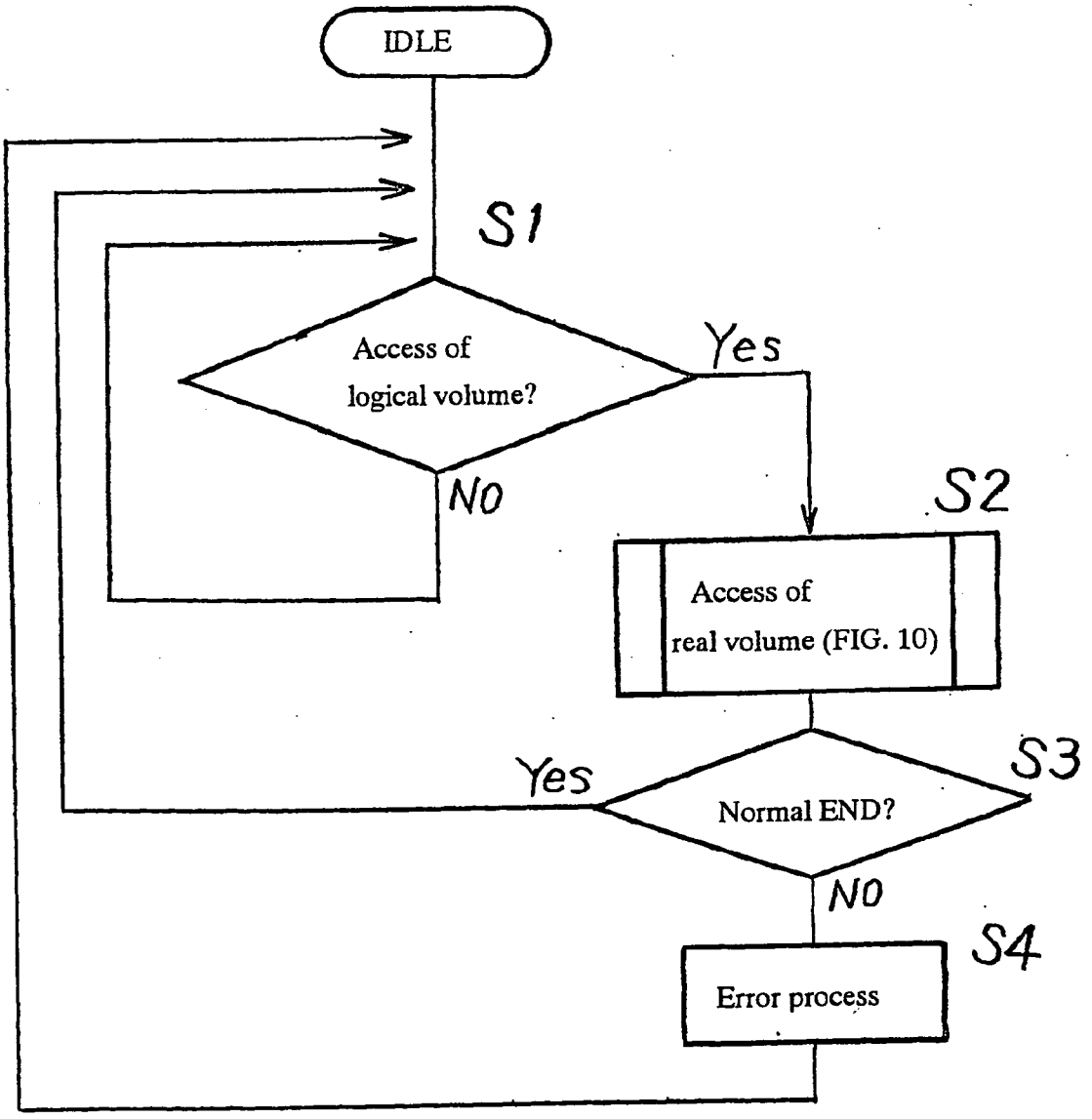

IDLE
S1
Access of
logical volume?
Yes
NO
S2
Access of
real volume (FIG. 10)
S3
Yes
Normal END?
NO
S4
Error process

FIG. 10

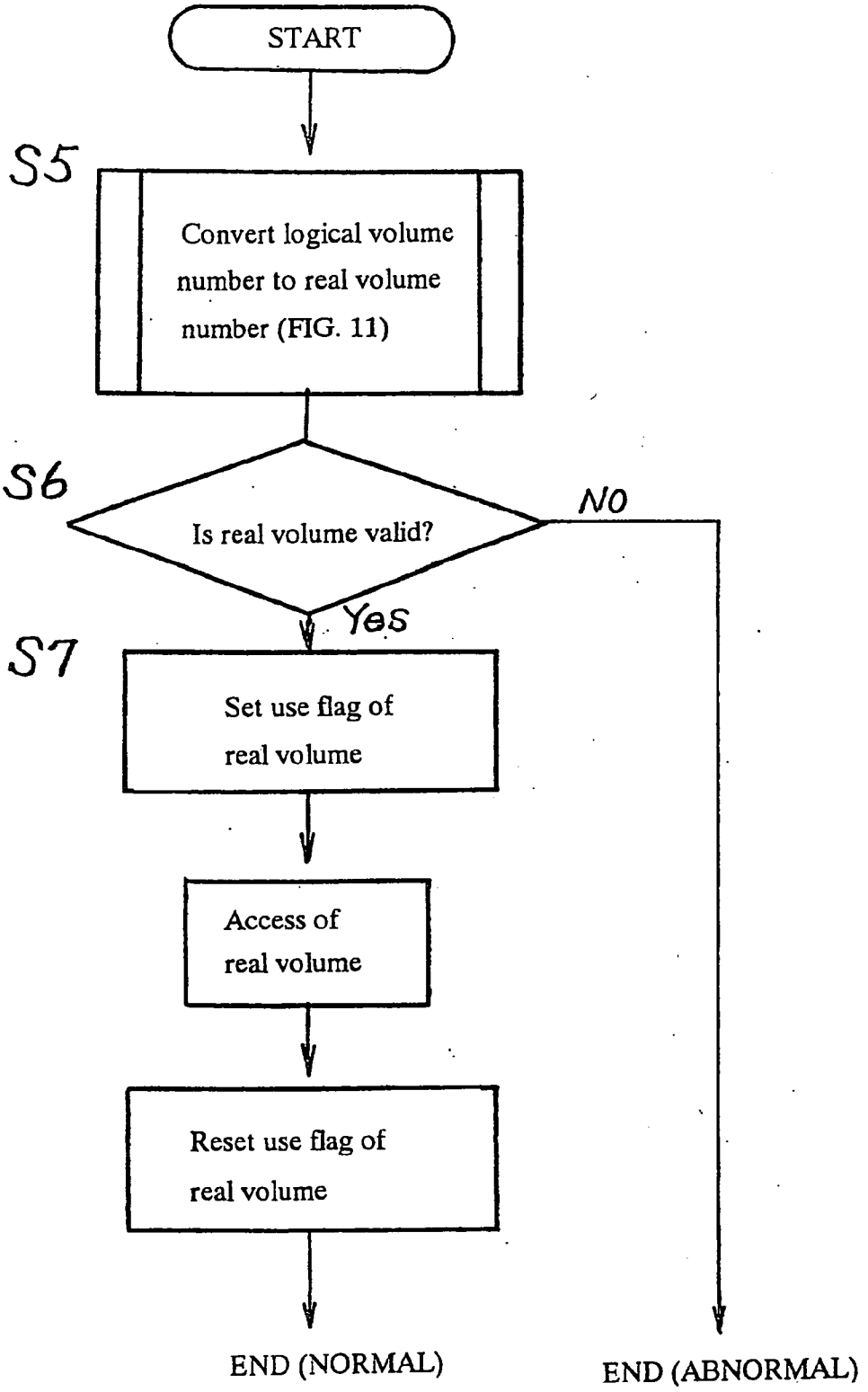

START
S5
Convert logical volume number to real volume number (FIG. 11)
S6
Is real volume valid?
NO
Yes
S7
Set use flag of real volume
Access of real volume
Reset use flag of real volume
END (NORMAL)
END (ABNORMAL)

FIG. 11

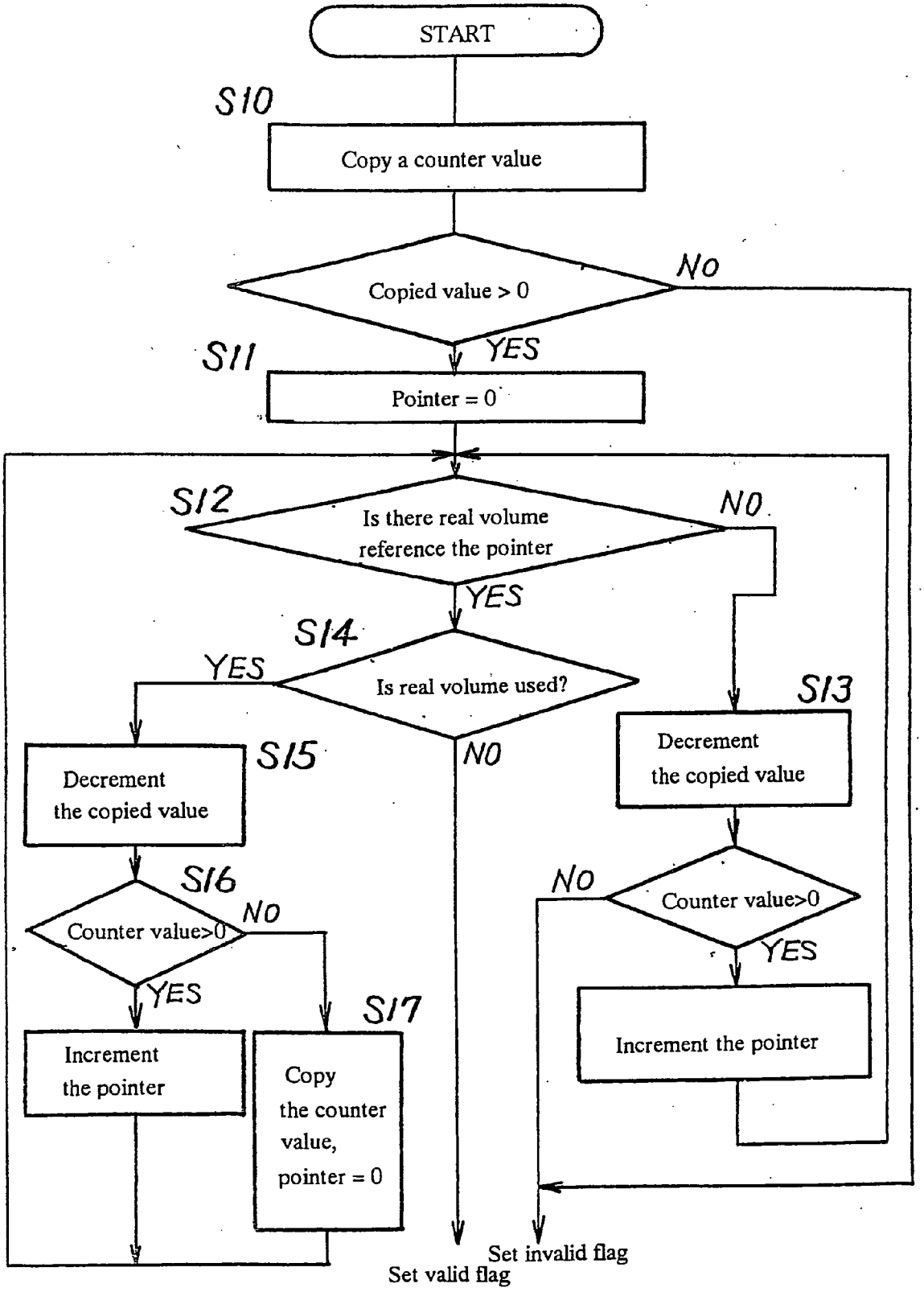

START
S10
Copy a counter value
Copied value > 0
No
YES
S11
Pointer = 0
S12
Is there real volume reference the pointer
NO
YES
S14
YES
Is real volume used?
S13
Decrement the copied value
NO
S15
Decrement the copied value
S16
Counter value>0
NO
YES
S17
Increment the pointer
Copy the counter value, pointer = 0
No
Counter value>0
YES
Increment the pointer
Set invalid flag
Set valid flag

FIG. 12

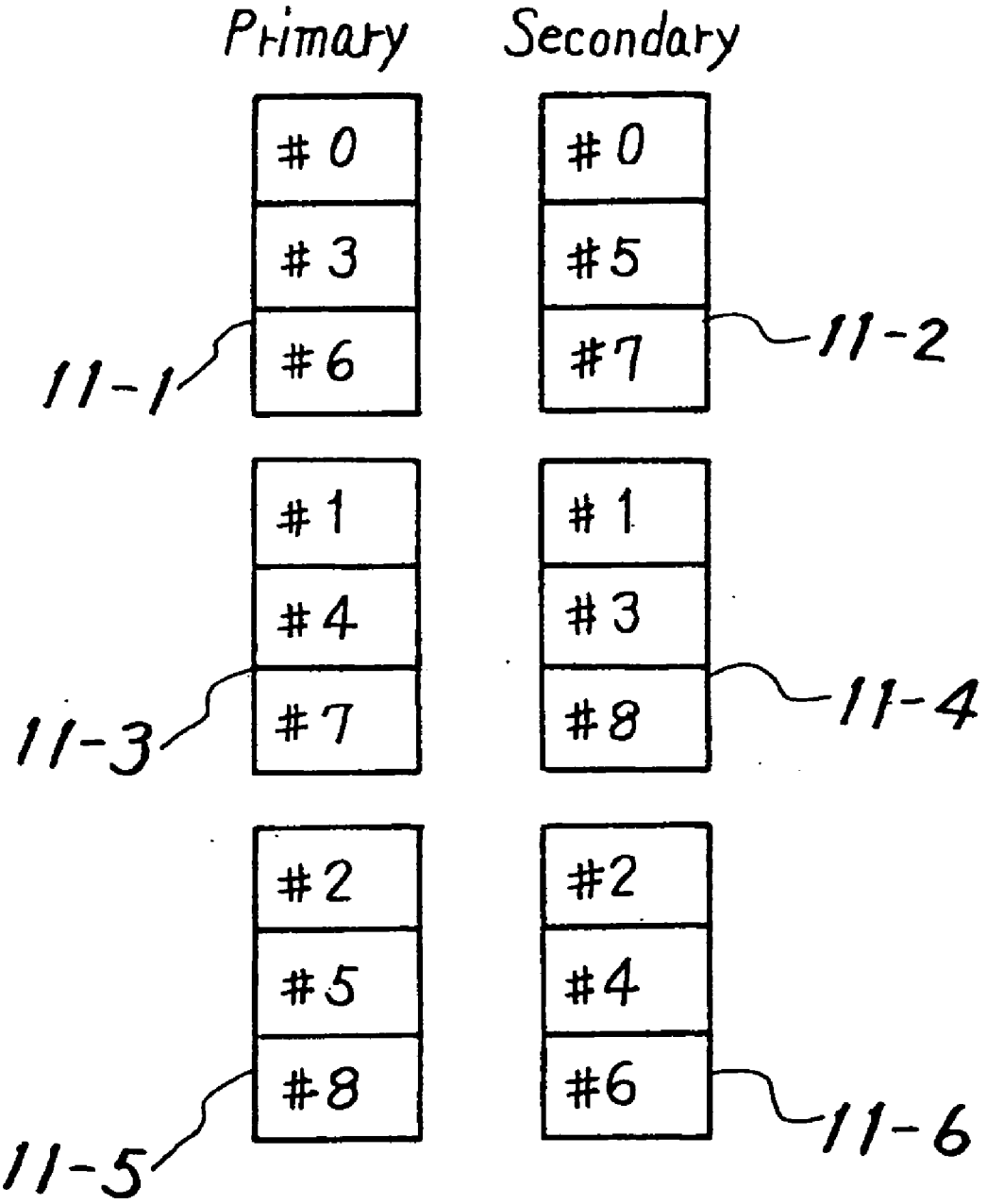

Primary
Secondary
#0
#3
#6
#0
#5
#7
11-1
11-2
#1
#4
#7
#1
#3
#8
11-3
11-4
#2
#5
#8
#2
#4
#6
11-5
11-6

FIG. 13

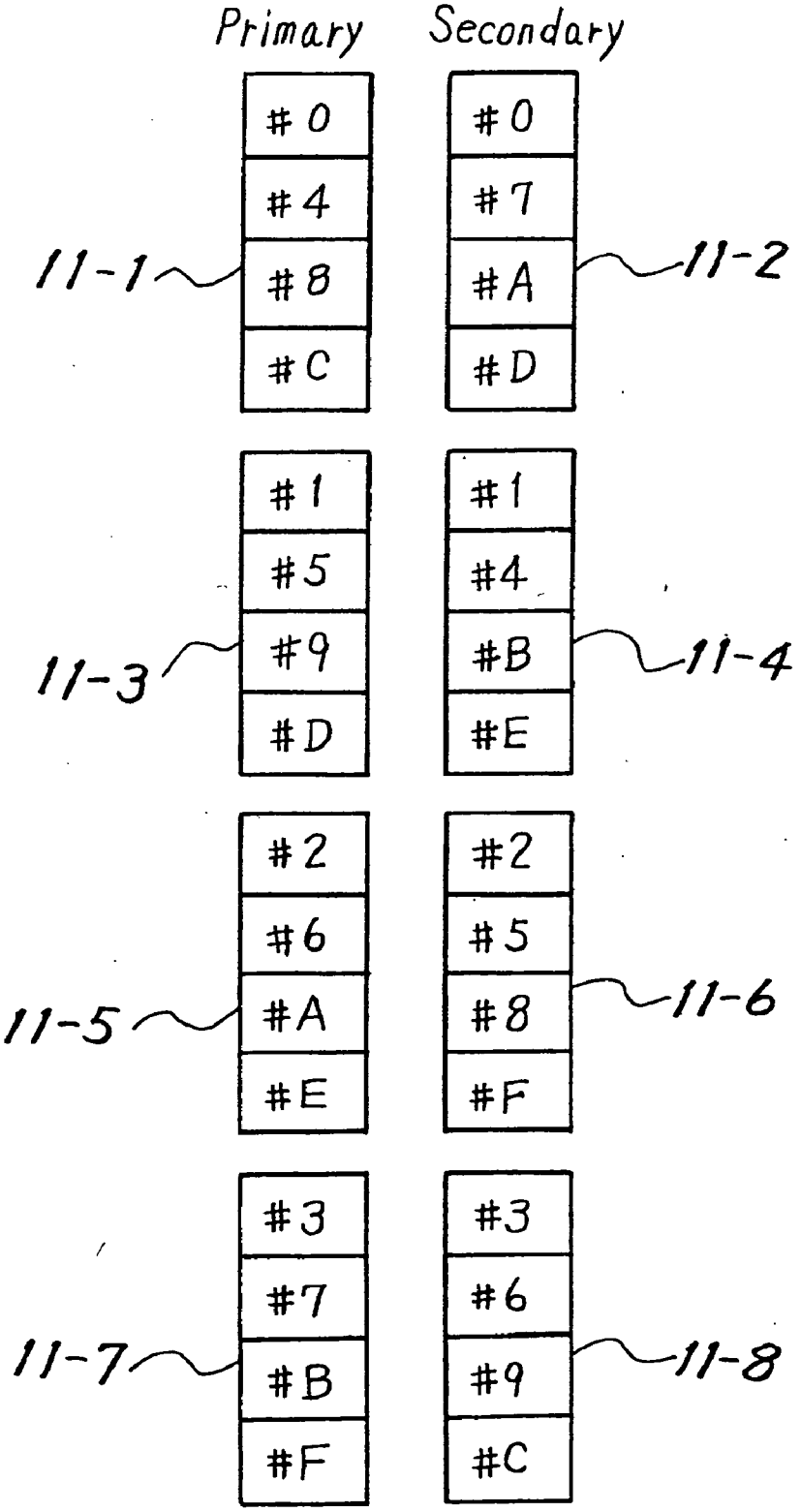

Primary
Secondary
#0
#4
#8
#C
#0
#7
#A
#D
11-1
11-2
#1
#5
#9
#D
#1
#4
#B
#E
11-3
11-4
#2
#6
#A
#E
#2
#5
#8
#F
11-5
11-6
#3
#7
#B
#F
#3
#6
#9
#C
11-7
11-8

FIG. 14

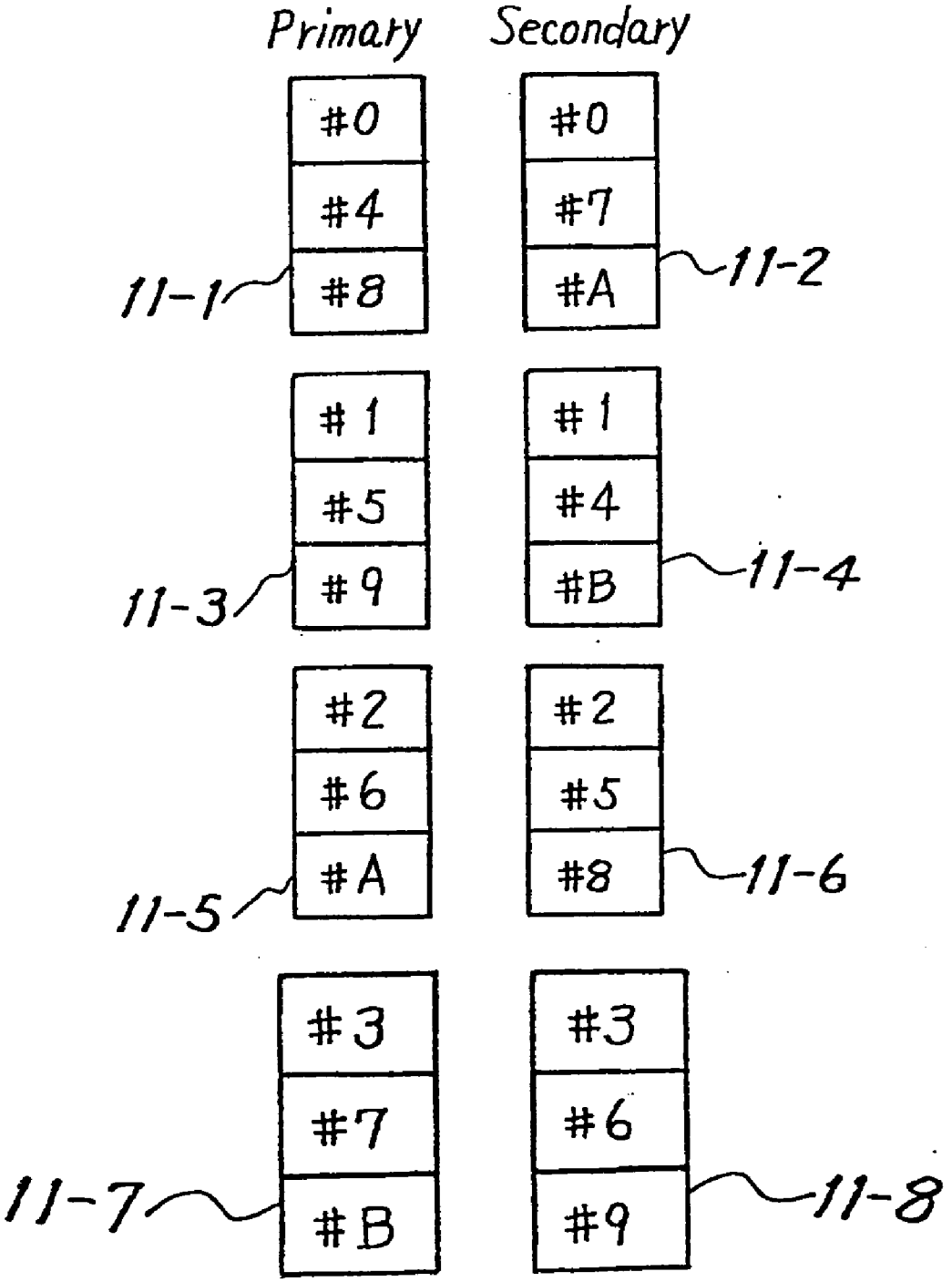
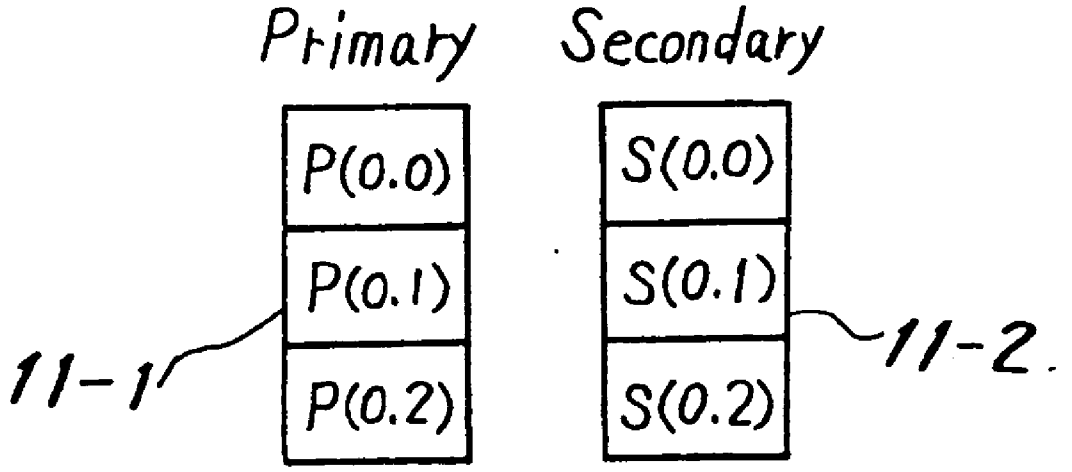

Primary
Secondary
#0
#4
#8
#0
#7
#A
11-1
11-2
#1
#5
#9
#1
#4
#B
11-3
11-4
#2
#6
#A
#2
#5
#8
11-5
11-6
#3
#7
#B
#3
#6
#9
11-7
11-8

FIG. 15



$$P(i,j)=i+n-j$$

$$S(i,j)=\underbrace{P(i-j,j)}_{(i-j\geq 0)} \text{ or } \underbrace{P(n+i-j,j)}_{(i-j<0)}$$

$$i=0\sim m-1,\ j=0\sim n-1$$

FIG. 16

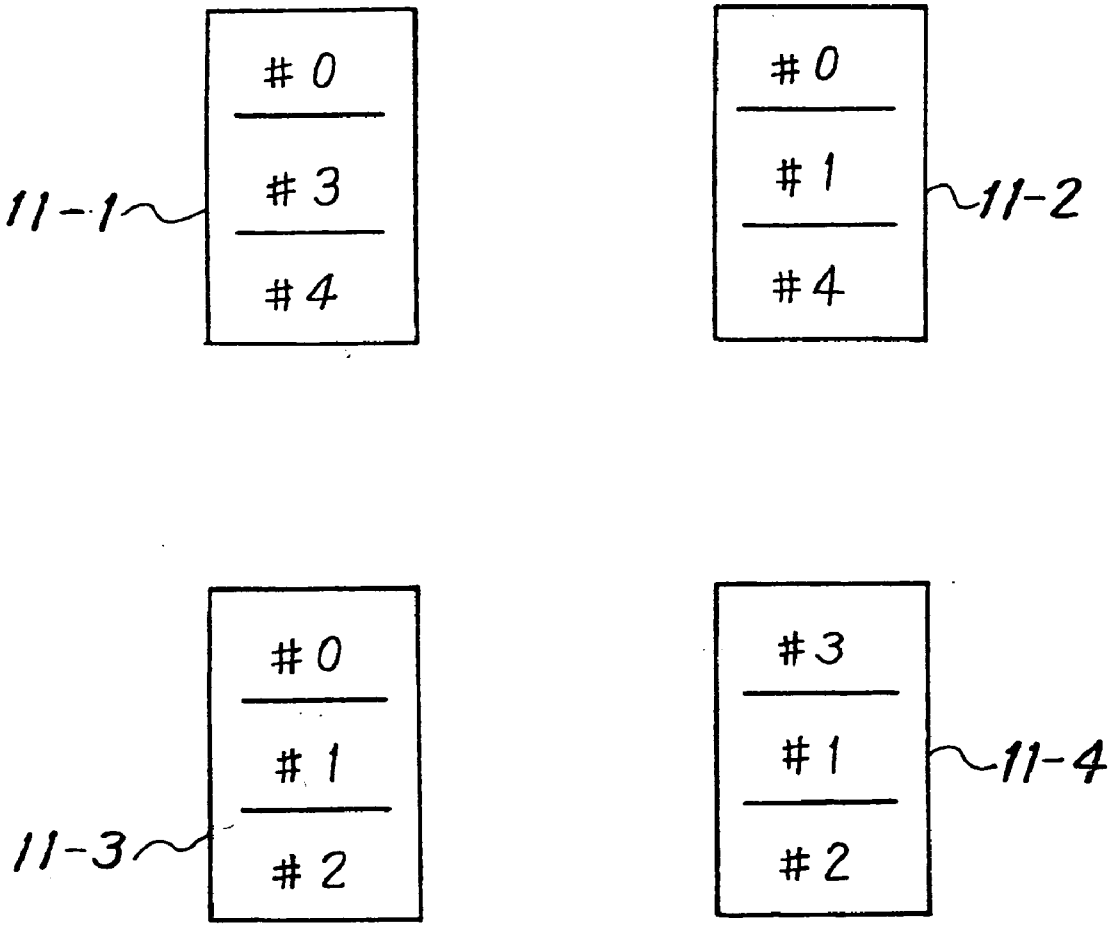


FIG. 17

PRIOR ART

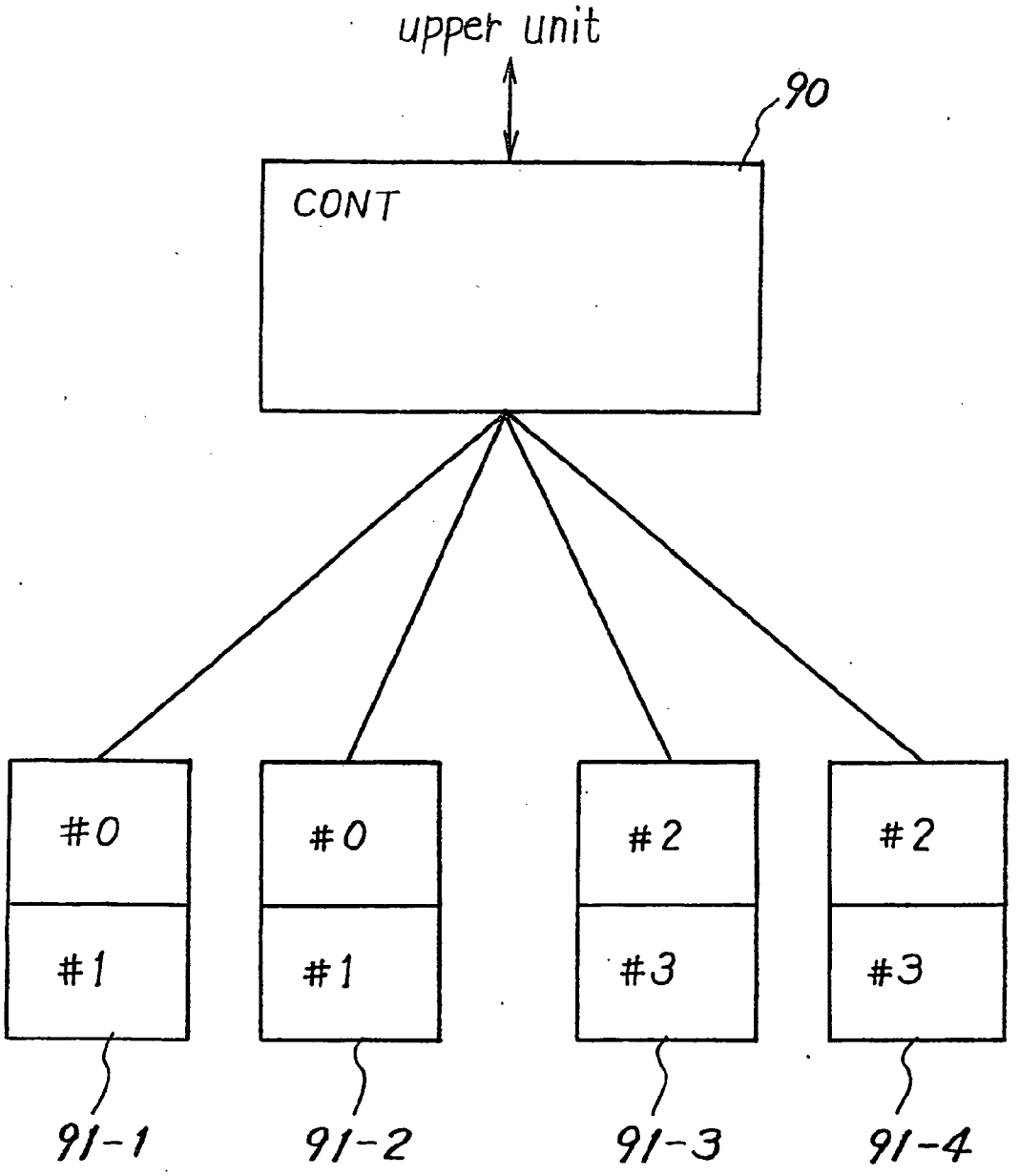


FIG. 18A

PRIOR ART

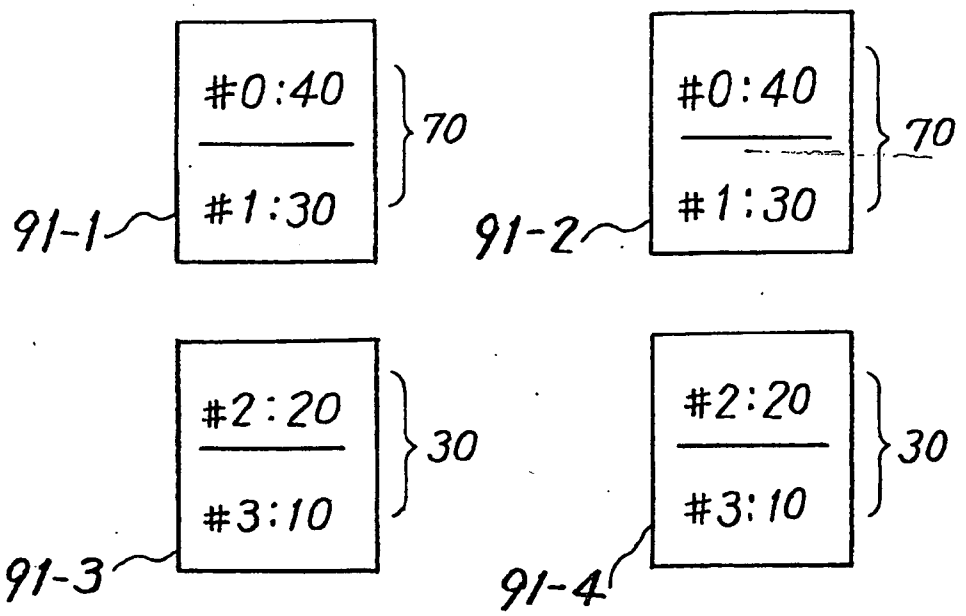


FIG. 18B

PRIOR ART

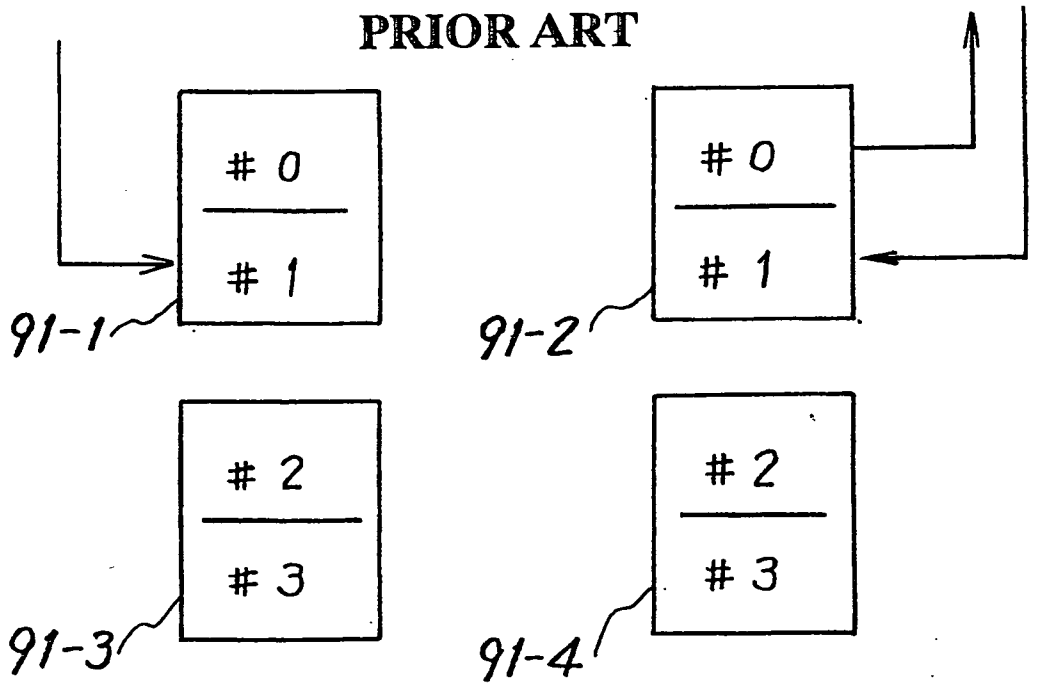
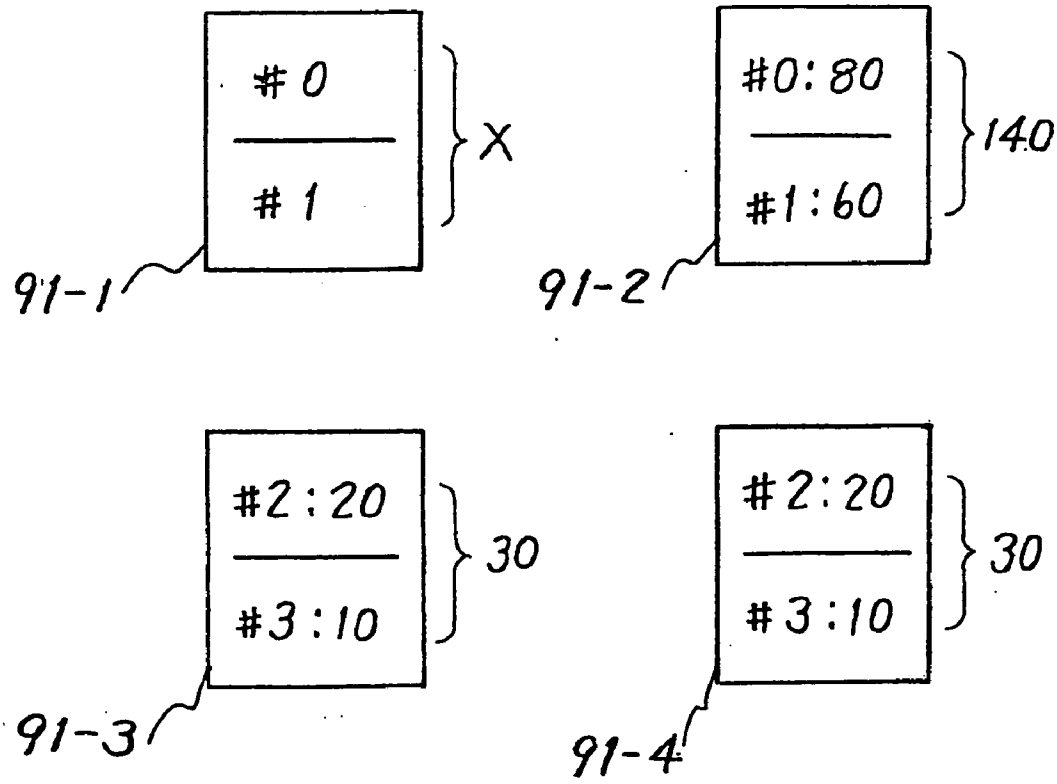


FIG. 19

PRIOR ART



RAID APPARATUS STORING A PLURALITY OF SAME LOGICAL VOLUMES ON DIFFERENT DISK UNITS

BACKGROUND OF THE INVENTION

[0001] 1. Field of the Invention

[0002] The present invention relates to a RAID (Redundant Arrays of Inexpensive Disks) apparatus which allocates a plurality of same logical volumes to a real volume consisting of a plurality of physical disk units, and, more particularly, to a RAID apparatus which efficiently uses physical disk units with a plurality of logical volumes.

[0003] 2. Description of the Related Art

[0004] Disk systems like a magnetic disk system are used as an external storage system in a computer system. A host computer accesses such a disk system with a logical volume name that an OS (Operating System) recognizes. Logical volumes, which are units of disk devices the host computer recognizes, are allocated in a disk system.

[0005] If each logical volume is allocated just one in such a disk system, when a physical disk unit where that logical volume is located fails, the logical volume cannot be used any more.

[0006] To prevent this shortcoming, a RAID apparatus has been proposed. A RAID apparatus has a plurality of same logical volumes allocated on different disk units. Even when one disk unit fails, another disk unit where the same logical volume is allocated can be used to access that logical volume.

[0007] **FIG. 17** is a structural diagram of prior art. As shown in **FIG. 17**, a RAID apparatus comprises a plurality of magnetic disk units **91-1** to **91-4** and a disk controller **90** which controls those disk units. The capacity of each of the magnetic disk units **91-1** to **91-4** has been increased drastically. There is a limit to the capacity of a volume that an OS recognizes. For instance, while a single magnetic disk unit has a capacity of 9 gigabytes, a volume recognizable by an OS is a maximum of about 4.5 gigabytes.

[0008] In this respect, each of the magnetic disk units (real volumes) **91-1** to **91-4** is divided to volumes recognizable by an OS (which are called "logical volumes").

[0009] The conventional scheme of combining logical volumes on the real volumes **91-1** to **91-4** was to make the combination of logical volumes on each real volume identical to the combination of logical volumes on another associated real volume.

[0010] As shown in **FIG. 17**, for example, the mirror structure of RAID-1 allocates two different logical volumes to each of the four magnetic disk units **91-1** to **91-4**. Specifically, logical volumes #0 and #1 are allocated on the magnetic disk units **91-1** and **91-2**, and logical volumes #2 and #3 on the magnetic disk units **91-3** and **91-4**. This scheme maintains the contents of two real volumes identical to each other.

SUMMARY OF THE INVENTION

[0011] **FIGS. 18A, 18B** and **19** are diagrams for explaining the problems of the prior art. The loads on the individual logical volumes differ from one another depending on the properties of the logical volumes. Suppose that the numbers of accesses to the logical volumes #0, #1, #2 and #3 are 80 times, 60 times, 40 times and 20 times, respectively. **FIG. 18A** shows the corresponding numbers of accesses to the physical disk units in the conventional allocation of the logical volumes. Since the logical volumes #0 and #1 are allocated on the physical disk unit **91-1**, the number of accesses becomes 70 times. Likewise, as the logical volumes #0 and #1 are also allocated on the physical disk unit **91-2**, the number of accesses becomes 70 times. Because the logical volumes #2 and #3 are allocated on the physical disk unit **91-3**, the number of accesses becomes 30 times. Likewise, as the logical volumes #2 and #3 are also allocated on the physical disk unit **91-4**, the number of accesses becomes 30 times.

[0012] That is, uneven loads on the logical volumes directly result in uneven loads on the real volumes. For example, the loads on the physical disk units **91-1** and **91-2** where the logical volumes #0 and #1 are allocated become greater, while the loads on the physical disk units **91-3** and **91-4** where the logical volumes #2 and #3 are allocated become smaller.

[0013] Therefore, uneven loads on the logical volumes lead to uneven loads on the physical disk units. This causes a problem of increasing the chance of high-load physical disk units being busy, thus reducing the overall access speed.

[0014] In some cases, sequential copy may be made between logical volumes. For example, copying from the logical volume #0 to the logical volume #1 may be made as shown in **FIG. 18B**. In this example, contention to the physical disk unit **91-1** occurs when copying is done from the logical volume #0 of the physical disk unit **91-1** to the logical volume #1 of the physical disk unit **91-1**.

[0015] Likewise, when copying is done from the logical volume #0 of the physical disk unit **91-2** to the logical volume #1 of the physical disk unit **91-1**, access to the logical volume #1 of the physical disk unit **91-2** contends the copying operation. If sequential copy is made between logical volumes, therefore, the performance is considerably reduced.

[0016] Further, a physical disk unit may be damaged so that it should be in retreat mode. Suppose that the physical disk unit **91-1** fails as shown in **FIG. 19**. According to the prior art, the logical volumes allocated on the physical disk unit **91-1** are also allocated on the physical disk unit **91-2**, so that loads will be concentrated on the physical disk unit **91-2**.

[0017] If the access numbers of the individual logical volumes are as exemplified in **FIG. 18A**, with the physical disk unit **91-1** failing, the access numbers of the physical disk units **91-3** and **91-4** do not change, while the access number of the physical disk unit **91-2** alone increases, as shown in **FIG. 19**.

[0018] In retreat mode, therefore, loads are undesirably concentrated on the physical disk unit where the same logical volumes allocated on the failing physical disk unit are allocated, thus lowering the performance.

[0019] Accordingly, it is an object of the present invention to provide a RAID apparatus which prevents loads from being imparted on real volumes due to uneven loads on logical volumes.

[0020] It is another object of this invention to provide a RAID apparatus which auto-adjusts loads between real volumes.

[0021] It is a further object of this invention to provide a RAID apparatus which prevents the performance from being reduced by uneven loads on logical volumes.

[0022] A RAID apparatus according to this invention comprises a real volume including a plurality of physical disk units on each of which a plurality of different logical volumes are allocated, and a disk controller for accessing any physical disk unit where a designated logical volume is allocated in order to access the designated logical volume

[0023] The real volume is designed in such a manner that a plurality of same logical volumes are respectively allocated on different physical disk units and a combination of a plurality of logical volumes allocated on each physical disk unit differs from one physical disk unit to another.

[0024] As apparent from the above, the combination of logical volumes to be allocated differs from one physical disk unit to another. Even if loads are not evenly imparted on the logical volumes, therefore, loading on the physical disk units is adjusted because of the difference in the combinations of the logical volumes to be allocated between the physical disk units. It is thus possible to prevent uneven loading on the physical disk units from occurring due to uneven loads on the logical volumes.

[0025] In a process of making a copy from one logical volume to another, it is possible to prevent contention of access to logical volumes which are undergoing the copying process, due to the difference in the combinations of the logical volumes to be allocated between the physical disk units.

[0026] Even in retreat mode, loads are distributed to the whole physical disk units to prevent over-concentration of loads for the combination of the logical volumes to be allocated on each physical disk unit differs from one physical disk unit to another.

[0027] Other features and advantages of the present invention will become readily apparent from the following description taken in conjucntion with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0028] The accompanying drawings, which are incorporated in and constitute a part of the specification, illustrate presently preferred embodiments of the invention, and together with the general description given above and the detailed description of the preferred embodiments given below, serve to explain the principle of the invention.

[0029] **FIG. 1** is a principle diagram of this invention;

[0030] **FIG. 2** is a structural diagram of one embodiment of this invention;

[0031] **FIG. 3** is an explanatory diagram of a disk unit according to the embodiment in **FIG. 2**;

[0032] **FIG. 4** is an explanatory diagram of an automatic load adjusting operation according to the embodiment in **FIG. 2**;

[0033] **FIG. 5** is an explanatory diagram of the operation of the embodiment in **FIG. 2** at the time copying is performed;

[0034] **FIG. 6** is an explanatory diagram of the operation of the embodiment in **FIG. 2** in retreat mode;

[0035] **FIG. 7** is an explanatory diagram of a converting mechanism according to the embodiment in **FIG. 2**;

[0036] **FIGS. 8A and 8B** are structural diagrams of the converting mechanism in **FIG. 7**;

[0037] **FIG. 9** is a flowchart illustrating an access process by the structure in **FIG. 7**;

[0038] **FIG. 10** is a flowchart showing a process of accessing a real volume in **FIG. 9**;

[0039] **FIG. 11** is a flowchart illustrating a process of accessing a real volume number in **FIG. 10**;

[0040] **FIG. 12** is an explanatory diagram of the second embodiment of this invention;

[0041] **FIG. 13** is an explanatory diagram of the third embodiment of this invention;

[0042] **FIG. 14** is an explanatory diagram of the fourth embodiment of this invention;

[0043] **FIG. 15** is an explanatory diagram of the fifth embodiment of this invention;

[0044] **FIG. 16** is an explanatory diagram of the sixth embodiment of this invention;

[0045] **FIG. 17** is a structural diagram of prior art;

[0046] **FIGS. 18A and 18B** are explanatory diagrams (part 1) of conventional problems; and

[0047] **FIG. 19** is an explanatory diagram (part 2) of conventional problems.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0048] **FIG. 1** presents a principle diagram of this invention.

[0049] As shown in **FIG. 1**, real volumes have a plurality of physical disk units **11-1** to **11-4** on which a plurality of different logical volumes are allocated. A disk controller **10** accesses any of the physical disk units **11-1** to **11-4** where a designated logical volume is allocated, in order to access this designated logical volume.

[0050] A plurality of same logical volumes are respectively allocated on different physical disk units. Further, the combination of a plurality of logical volumes allocated on each physical disk unit is made different from one physical disk unit to another.

[0051] For example, logical volumes #0 and #1 are allocated on the physical disk unit **11-1**. Logical volumes #0 and #3 are allocated on the physical disk unit **11-2**. Logical volumes #2 and #3 are allocated on the physical disk unit **11-3**. Logical volumes #2 and #1 are allocated on the physical disk unit **11-4**. The real volume is therefore designed in such a way that the combination of a plurality of logical volumes allocated on each physical disk unit differs from one physical disk unit to another.

[0052] **FIG. 2** is a structural diagram of one embodiment of this invention, **FIG. 3** is an explanatory diagram of a disk unit according to this embodiment, **FIG. 4** is an explanatory diagram of an automatic load adjusting operation in **FIG. 3**, **FIG. 5** is an explanatory operational diagram at the time copying with the disk unit in **FIG. 3** is performed, and **FIG. 6** is an explanatory diagram of the operation of the disk unit in **FIG. 3** in retreat mode.

[0053] Referring to **FIG. 2**, the disk controller **10** is constituted of a magnetic disk controller. The disk controller **10** has a channel adapter **20** for exchanging commands/data with a high-rank apparatus like a computer, a resource manager **21** for managing resources, a table storage **22** for storing various kinds of tables, a main storage **23** for storing read/write data, etc., and two device adapters **24** and **25** for controlling the devices (magnetic disk units) **11-1** to **11-4**.

[0054] The magnetic disk units **11-1** to **11-4**, each constituted of a known magnetic disk storage device, are connected to the associated device adapters **24** and **25**.

[0055] In the magnetic disk units **11-1** to **11-4**, logical volumes are allocated as shown in **FIG. 3**. The following description will given of the case where the RAID structure is the mirror structure of RAID-**1**. The mirror structure has each of the logical volumes #0 to #3 allocated double. That is, the same logical volume is allocated double.

[0056] In this example, it is assumed that two logical volumes are allocated on each of the magnetic disk units **11-1** to **11-4**. The logical volumes #0 and #1 are allocated on the magnetic disk unit **11-1**. The logical volumes #0 and #3 are allocated on the magnetic disk unit **11-2**. The logical volumes #2 and #3 are allocated on the magnetic disk unit **11-3**. The logical volumes #2 and #1 are allocated on the magnetic disk unit **11-4**.

[0057] Therefore, the same logical volume is allocated on different magnetic disk units, and the combination of logical volumes allocated on each magnetic disk unit differs from one magnetic disk unit to another.

[0058] The operation of a system which has logical volumes allocated in asymmetrical fashion will now be described.

[0059] Suppose that the numbers of accesses to the logical volumes #0, #1, #2 and #3 are 80 times, 60 times, 40 times and 20 times, respectively, as shown in **FIG. 4**. Since the logical volumes #0 and #1 are allocated on the magnetic disk unit **11-1** in the asymmetrical allocation, the access number becomes 70 times, as shown on the left-hand side in **FIG. 4**. Likewise, as the logical volumes #0 and #3 are also allocated on the magnetic disk unit **11-2**, the access number becomes 50 times.

[0060] Because the logical volumes #2 and #3 are allocated on the magnetic disk unit **11-3**, the access number becomes 30 times. Likewise, as the logical volumes #2 and #1 are also allocated on the magnetic disk unit **11-4**, the access number becomes 50 times.

[0061] In the normal operation, to access a logical volume, a magnetic disk unit which is not in use is accessed. As the access operation proceeds, therefore, the loads are gradually adjusted to the state shown on the right-hand side in **FIG. 4**.

[0062] The logical volumes #1 and #3 which are respectively allocated on the magnetic disk units **11-1** and **11-2** which both include the logical volume #0 have their pairs in the magnetic disk units **11-3** and **11-4** which do not include the logical volume #0. The frequency of processing on the logical volumes #1 and #3 (the read requesting process) performed on the magnetic disk units **11-3** and **11-4** which do not include the logical volume #0 that has a relatively low load increases.

[0063] The adjustment of the loads is executed in a chain-like fashion. At last, load adjustment is performed over the entire magnetic disk units **11-1** to **11-4** in the group. Accordingly, the high load on the magnetic disk unit **11-1** is distributed to the magnetic disk units **11-2**, **11-3** and **11-4** which have low loads.

[0064] The access numbers to the individual magnetic disk units **11-1** to **11-4** all become equal to 50 times. Even if there are uneven loads on the logical volumes, therefore, the loads on the magnetic disk units become even. Even if there are uneven loads on the logical volumes, thus, the response speeds of the individual magnetic disk units become identical.

[0065] This can shorten the standby time for access to even a high-load logical volume. That is, the access speed of the disk system having a mirror structure can be improved.

[0066] **FIG. 5** illustrates the operation of copying from the logical volume #0 to the logical volume #1. In this example, the logical volume #0 on the magnetic disk unit **11-2** is copied on the logical volume #1 on the magnetic disk unit **11-1**.

[0067] In this case, even when there is an access to the logical volume #1, this logical volume #1 is allocated on the magnetic disk unit **11-4** which is free, so that the logical volume #1 can be accessed by accessing the magnetic disk unit **11-4**.

[0068] The same is true of copying between other logical volumes. As access contention to the logical volumes can be avoided even when copying between the logical volumes is in progress, the response performance is improved.

[0069] **FIG. 6** is an explanatory diagram of the operation in retreat mode. The left-hand state in **FIG. 6** shows the state after the adjustment shown on the right-hand side in **FIG. 4**. When one magnetic disk unit **11-1** fails in this case, the other magnetic disk units **11-2** to **11-4** are used.

[0070] In this case, the load on the failing magnetic disk unit **11-1** is distributed to the magnetic disk units **11-2** to **11-4** which are not malfunctioning. The access number to the magnetic disk unit **11-2** is 80 times, the access number to the magnetic disk unit **11-3** is 60 times, and the access number to the magnetic disk unit **11-4** is 60 times.

[0071] In general, in retreat mode where one of each pair of magnetic disk units in the mirror structure fails and cannot be used, the performance of the read requesting process becomes halved according to the conventional structure. According to this invention, by contrast, the performance of the read requesting process would be reduced merely by 25% due to the load distribution.

[0072] Note that the logical volumes #0 to #3 are located at the same positions on the magnetic disk units **11-1** to **11-4**, as shown in **FIG. 3**. For instance, the logical volume #1 is located on the lower side of the magnetic disk units **11-1** and **11-4**.

[0073] In the block indicating each magnetic disk unit in this diagram, the upper side is the outer peripheral side of a magnetic disk and the lower side is the inner peripheral side of the magnetic disk. In other words, the logical volume #1 is allocated on the inner peripheral side on the magnetic disk units **11-1** and **11-4**.

[0074] A description will now be given of the reason why a plurality of same logical volumes are allocated on the magnetic disk units at the same position. A magnetic disk unit has different amounts of data per track between the inner peripheral side and the outer peripheral side. The amount of data to be read/written per track or per unit time is greater on the outer peripheral side. This means that the closer to the outer periphery of the magnetic disk the access point is located, the faster the read/write time per data becomes.

[0075] With the use of the mirror structure, writing is simultaneously performed on a pair of logical volumes. If the locations of the logical volumes vary on the magnetic disks, therefore, the write time varies. It is necessary to tune the writing operation to the one having a slower writing speed, which deteriorates the overall performance.

[0076] When a plurality of same logical volumes are allocated on magnetic disks at the same position, on the other hand, the write time becomes the same between those same logical volumes. The read/write time in the mirror structure is thus improved.

[0077] The structure and operation will be discussed more specifically. **FIG. 7** is an explanatory diagram of a converting mechanism between logical volumes in **FIG. 2**, and **FIGS. 8A and 8B** are structural diagrams of the converting mechanism in **FIG. 7**.

[0078] As shown in **FIG. 7**, queued accesses to the individual logical volumes #0 to #3 are converted to accesses to the magnetic disk units (real volumes) **11-1** to **11-4** by the resource manager **21** and the table storage **22**.

[0079] Provided in the table storage **22** are a conversion table **22-1** shown in **FIG. 8A** and a use-state table **22-2** shown in **FIG. 8B**.

[0080] The conversion table **22-1** stores, for each of the logical volumes #0-#3, the number of the real volume (magnetic disk unit) where the associated logical volume is allocated. For each logical volume, the conversion table **22-1** further stores a counter representing the multiplicity that indicates the quantity of that logical volume.

[0081] In the example in **FIG. 3**, for instance, the numbers of the magnetic disk units **11-1** and **11-2** are set as the numbers of the real volumes with respect to the logical volume #0. As two logical volumes #0 are allocated, the counter becomes "2."

[0082] For a failing real volume or a real volume which does not exist, an invalid flag is set in the associated column of the real volume number.

[0083] The use-state table **22-2** indicates whether or not each real volume (magnetic disk unit) is in use, by means of a use flag as shown in **FIG. 8B**.

[0084] **FIG. 9** is a flowchart illustrating an access process according to the embodiment in **FIG. 2**, **FIG. 10** is a flowchart showing a process of accessing a real volume in **FIG. 9**, and **FIG. 11** is a flowchart illustrating a process of accessing a real volume number in **FIG. 10**.

[0085] The access process in **FIG. 9** will be discussed below.

[0086] (S1) The resource manager **21** is constituted of a processor. The resource manager (hereinafter called "processor") **21** checks if there is an access to a logical volume by the channel adapter **20**.

[0087] (S2) When determining that there is such an access, the processor **21** executes the process of accessing a real volume illustrated in **FIG. 10**.

[0088] (S3) The processor **21** determines if the accessing has been terminated properly. If the decision is the proper access termination, the processor **21** returns to step S1.

[0089] (S4) If the access has not been terminated properly, the processor **21** performs an error process and then returns to step S1. The proceess of accessing a real value in **FIG. 10** will now be discussed.

[0090] (S5) When determining that there is an access, the processor **21** acquires the number of the real volume to access data from the logical volume number in the process which will be discussed later with reference to **FIG. 11**.

[0091] (S6) When obtaining the real volume number, the processor **21** checks if the real volume is valid. When determining that the real volume is invalid, the processor **21** sets an abnormal termination flag and returns to the routine in **FIG. 9**.

[0092] (S7) When determining that the real volume is valid, the processor **21** sets the use flag for this real volume in the use-state table **22-2** in the table storage **22**. Then, the processor **21** accesses that real volume. That is, the processor **21** accesses the real volume (magnetic disk unit) via the device adapters **24** and **25**. After the access is completed, the processor **21** erases the use flag for the real volume in the use-state table **22-2**. The processor **21** further sets the normal termination flag and returns to the routine in **FIG. 9**.

[0093] Step S3 in **FIG. 9** is for checking the normal termination flag or the abnormal termination flag to determine whether the process in **FIG. 10** has been terminated normally or abnormally.

[0094] The process of accessing a real volume number in **FIG. 11** will now be discussed.

[0095] (S10) The processor **21** copies the count value of the logical volume in question from the conversion table **22-1** (see **FIG. 8A**) in the table storage **22**. The processor **21** determines if the copied count value exceeds "0." As the count value indicates the multiplicity of an allocated logical volume, the copied count value of "0" means that the logical volume has not been allocated. Unless the copied count value is greater than "0," therefore, the processor **21** sets the invalid flag indicating invalidness and then returns to the routine in **FIG. 10**.

[0096] (S11) When the copied count value is greater than "0," the processor **21** sets the pointer for addressing the real volume number in the conversion table **22-1** to "0."

[0097] (S12) Then, the processor **21** refers to the real volume number column in the conversion table **22-1** which

is indicated by the pointer. The processor **21** checks if a real volume number is in the referred real volume number column.

[**0098**] (S**13**) When a real volume number is not present in the referred real volume number column, the processor **21** decrements the copied count value by "1." Then, the processor **21** determines if the copied count value exceeds "0." When the copied count value does not exceed "0," which means that the logical volume has not been allocated, the processor **21** sets the invalid flag indicating invalidness and then returns to the routine in **FIG. 10** as done in the above-described step S**10**. When the copied count value is greater than "0," on the other hand, the processor **21** increments the pointer for addressing the real volume number in the conversion table **22-1** by "1" and then returns to step S**12**.

[**0099**] (S**14**) When determining in step S**12** that the real volume indicated by the pointer exists, the processor **21** refers to the use-state table **22-2** to determine if that real volume is currently in use. When this real volume is not in use, the processor **21** sets a valid flag and returns to the routine in **FIG. 10**.

[**0100**] (S**15**) When determining in step S**14** that the real volume is in use, the processor **21** searches for another real volume. For this purpose, the processor **21** decrements the copied count value by "1."

[**0101**] (S**16**) The processor **21** determines if this count value exceeds "0." When the copied count value exceeds "0," the processor **21** increments the pointer for addressing the real volume number in the conversion table **22-1** by "1" and then returns to step S**12**.

[**0102**] (S**17**) When the copied count value does not exceed "0," on the other hand, there is no further real volume on which logical volumes have been allocated. Therefore, the processor **21** recopies the count value. Then, the processor **21** sets the pointer back to "0" and then returns to step S**12**.

[**0103**] By referring to the conversion table **22-1** and the use-state table **22-2**, the logical volumes are allocated and a real volume which is not being used is searched. Apparently, a real volume can easily searched by referring to those tables.

[**0104**] For any real volume which has become invalid, the invalid flag should be set. Even if there is an invalid real volume, searching for a free real volume can be accomplished without modifying the conversion table **22-1**.

[**0105**] A free real volume is searched by incrementing the pointer by the multiplicity set in the counter. Therefore, the multiplicities of the individual logical volumes can be set for each logical volume as will be shortly discussed in the description of other embodiments.

[**0106**] An example of the allocation of logical volumes will now be discussed.

[**0107**] **FIG. 12** is an explanatory diagram of the second embodiment of this invention, **FIG. 13** is an explanatory diagram of the third embodiment of this invention, **FIG. 14** is an explanatory diagram of the fourth embodiment of this invention, and **FIG. 15** is an explanatory diagram of the fifth embodiment of this invention.

[**0108**] The following description is given of the case of the mirror structure having pairs of physical disk units. Assume that m logical volumes are allocated on a single physical disk unit and the number of pairs of physical disk units is n.

[**0109**] **FIG. 12** shows the case of m=n=3. Specifically, three logical volumes are allocated on a single physical disk unit and three pairs of physical disk units are provided in this example. As shown in **FIG. 12**, the combinations of logical volumes #0 to #8 allocated on the individual physical disk units **11-1** to **11-6** differ from one physical disk unit to another.

[**0110**] The logical volumes allocated on the same physical disk unit on the primary side are allocated different physical disk units on the secondary side. For example, the logical volumes #0, #3 and #6 allocated on the physical disk unit **11-1** are respectively allocated on the physical disk units **11-2**, **11-4** and **11-6**. The position of each logical volume on the physical disk unit (the inner peripheral/outer peripheral position) is the same on another physical disk unit as in the example in **FIG. 3**.

[**0111**] **FIG. 13** shows the case of m=n=4. Specifically, four logical volumes are allocated on a single physical disk unit and four pairs of physical disk units are provided in this example. As shown in **FIG. 13**, the combinations of logical volumes #0 to #F allocated on the individual physical disk units **11-1** to **11-8** differ from one physical disk unit to another.

[**0112**] The logical volumes allocated on the same physical disk unit on the primary side are allocated different physical disk units on the secondary side. For example, the logical volumes #0, #4, #8 and #C allocated on the physical disk unit **11-1** are respectively allocated on the physical disk units **11-2**, **11-4**, **11-6** and **11-8**.

[**0113**] The position of each logical volume on the physical disk unit (the inner peripheral/outer peripheral position) is the same on another physical disk unit as in the example in **FIG. 3**.

[**0114**] **FIG. 14** shows the case of m=3 and n=4. Specifically, three logical volumes are allocated on a single physical disk unit and four pairs of physical disk units are provided in this example. As shown in **FIG. 14**, the combinations of logical volumes #0 to #B allocated on the individual physical disk units **11-1** to **11-8** differ from one physical disk unit to another.

[**0115**] The logical volumes allocated on the same physical disk unit on the primary side are allocated different physical disk units on the secondary side. For example, the logical volumes #0, #4 and #8 allocated on the physical disk unit **11-1** are respectively allocated on the physical disk units **11-2**, **11-4** and **11-6**.

[**0116**] The position of each logical volume on the physical disk unit (the inner peripheral/outer peripheral position) is the same on another physical disk unit as in the example in **FIG. 3**.

[**0117**] **FIG. 15** is a diagram for explaining a general equation for the examples of the allocation of logical volumes shown in **FIGS. 12 through 14**.

[**0118**] In the figure, m is the number of logical volumes present in a single physical disk unit, and n is the number of

pairs of physical disk units which constitute the mirror structure. P(x, y) is the physical disk number of a logical volume y in a physical disk x which becomes the primary, and S(x, y) is the physical disk number of a logical volume y in a physical disk x which becomes the secondary.

[0119] With P(i, j)=i+n×j,

[0120] when i−j 0, S(i, j)=P(i−j, j).

[0121] When i−j<0, S(i, j)=P(n+i−j, j).

[0122] **FIG. 16** is an explanatory diagram of the sixth embodiment of this invention.

[0123] **FIG. 16** shows the case of m=3 and n=2. Specifically, three logical volumes are allocated on a single physical disk unit and two pairs of physical disk units are provided in this example. As shown in **FIG. 16**, the combinations of logical volumes #0 to #4 allocated on the individual physical disk units **11-1** to **11-4** differ from one physical disk unit to another.

[0124] The logical volumes allocated on the same physical disk unit on the primary side are allocated different physical disk units on the secondary side. Further, the logical volumes #0 and #1 are provided triple and the logical volumes #2 and #3 are provided double.

[0125] As apparent from the above, the multiplicity of each logical volume can be selected in accordance with the frequency of accesses.

[0126] Besides the above-described embodiments, this invention may be modified as follows.

[0127] (1) Although the foregoing description has been given of the RAID-**1** or the mirror structure which has double logical volumes, this invention may be adapted to the structure which has logical volumes provided triple.

[0128] (2) Although the physical disk units have been explained as magnetic disk units, optical disk units, magneto-optical disk units or the like may be used as well.

[0129] Although several embodiments of this invention have been described herein, various other modifications can be made within the scope and spirit of this invention, and the present examples and embodiment are to be considered as illustrative and not restrictive.

[0130] As apparent from the above, this invention has the following advantages.

[0131] (1) As the combination of logical volumes to be allocated differs from one physical disk unit to another, even if loads are not evenly put on the logical volumes, loading on the physical disk units is adjusted. It is therefore possible to prevent uneven loading on the physical disk units from occurring due to uneven loads on the logical volumes.

[0132] (2) Due to the difference in the combinations of the logical volumes to be allocated between the physical disk units, it is possible to prevent contention of access to logical volumes which are undergoing a process of making a copy between logical volumes.

[0133] (3) Because of the difference in the combinations of the logical volumes to be allocated between the physical disk units, loads are distributed to the whole physical disk units even in retreat mode to thereby prevent over-concentration of loads.

1-8. (canceled)

9. A RAID accessing method comprising:

receiving an access request from a host that recognizes a logical volume as one volume; and

accessing any of physical disk units on which a designated logical volume by said access request is allocated in order to access said designated logical volume, said physical disk units each asymmetrically storing a different combination of logical volumes than each of the other physical disk units, wherein plural logical volumes of the plurality of logical volumes are allocated to the single physical disk unit;

wherein said accessing comprises accessing another physical disk unit on which said designated logical volume when one physical disk unit on which said designated logical volume is unused.

10. A RAID apparatus comprising:

a real volume including a plurality of physical disk units onto each of which a plurality of logical volumes are asymmetrically allocated; and

a disk controller receiving an access request form a host that recognizes a logical volume as one volume and accessing any of the physical disk units on which a designated logical volume by said access request is allocated in order to access said designated logical volume,

wherein said real volume comprises a plurality of same logical volumes respectively allocated to different physical disk units to each physical disk unit, said combination differing from one physical disk unit to another,

and wherein plural logical volumes of the plurality of logical volumes are allocated to the single physical disk unit,

and wherein said disk controller accesses another physical disk unit on which said designated logical volume when one physical disk unit on which said designated logical volume is unused.

* * * * *