

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第3748631号  
(P3748631)

(45) 発行日 平成18年2月22日(2006.2.22)

(24) 登録日 平成17年12月9日(2005.12.9)

(51) Int. Cl.

G06F 3/06 (2006.01)

F I

G06F 3/06 540

G06F 3/06 301J

請求項の数 5 (全 15 頁)

<p>(21) 出願番号 特願平8-217533                  (22) 出願日 平成8年8月19日(1996.8.19)                  (65) 公開番号 特開平10-63441                  (43) 公開日 平成10年3月6日(1998.3.6)                  審査請求日 平成15年3月25日(2003.3.25)</p>	<p>(73) 特許権者 396004833                  株式会社エクシング                  愛知県名古屋市瑞穂区塩入町18番1号                  (73) 特許権者 000005267                  ブラザー工業株式会社                  愛知県名古屋市瑞穂区苗代町15番1号                  (74) 代理人 100082500                  弁理士 足立 勉                  (72) 発明者 森川 憲                  愛知県名古屋市中区錦3丁目10番33号                  株式会社エクシング内                    審査官 木村 貴俊                    (56) 参考文献 特開平07-141121 (JP, A)                  最終頁に続く</p>
---	---

(54) 【発明の名称】 ディスクアレイにおける論理記憶領域統合方法およびディスクアレイ装置

(57) 【特許請求の範囲】

【請求項1】

データを記憶するディスク装置1台以上によって構成されるドライブ列を複数と、論理記憶領域を設定する論理記憶領域設定手段と、該論理記憶領域を複数の論理ブロックに分割するブロック設定手段と、各論理ブロックに前記ドライブ列の物理ブロックを割り当てる際に前記論理記憶領域において互いに隣接する前記論理ブロックにはそれぞれ別の前記ドライブ列の物理ブロックを割り当てる割り当て手段と、前記複数のドライブ列を並列に動作させるドライブ制御手段とを備えるディスクアレイにおいて、ドライブ列が増設された際に、増設分のドライブ列の構成が既設のドライブ列の構成と同じであることを前提として、既設のドライブ列に対応する論理記憶領域と増設分のドライブ列に対応する論理記憶領域とを統合するための論理記憶領域統合方法であって、

10

前記ドライブ列が増設されると、前記既設分のドライブ列に対応する前記論理記憶領域を第1の論理記憶領域とし、該増設分のドライブ列に応じた第2の論理記憶領域と該第2の論理記憶領域とは別の第3の論理記憶領域を設定し、該第2の論理記憶領域を複数の論理ブロックに分割して、論理ブロック番号を前記第1の論理記憶領域と前記第2の論理記憶領域とで異なるように割り振ってから、

該第1および第2の論理記憶領域の未使用の論理ブロックをそれぞれが属していた論理記憶領域から分離させ、該分離させた論理ブロックを前記第3の論理記憶領域に属する論理ブロックとする併合ステップと、

前記第1および第2の論理記憶領域の論理ブロックに割り当てられている前記物理ブロ

20

ックのデータを前記第3の論理記憶領域の論理ブロックに割り当てられた前記物理ブロックに移動させて該論理ブロックを前記未使用の論理ブロックとするデータ移動ステップとを、

前記第1および第2の論理記憶領域に属していた前記論理ブロックがすべて前記第3の論理記憶領域に属するまで繰り返し実行し、前記第1および第2の論理記憶領域に属していた前記論理ブロックを前記第3の論理記憶領域に併合する毎に前記第1および第2の論理記憶領域の論理ブロック数を前記分離させたブロック数分減算し、前記第3の論理記憶領域の論理ブロック数を該併合されたブロック数分加算することを特徴とするディスクアレイにおける論理記憶領域統合方法。

【請求項2】

請求項1記載のディスクアレイにおける論理記憶領域統合方法において、

前記併合ステップを実行する前に前記第1の論理記憶領域に未使用の論理ブロックが存在しないときには、該第1の論理記憶領域のいずれかの論理ブロックに割り当てられている前記物理ブロックのデータを前記第2の論理記憶領域の未使用の論理ブロックに割り当てられている前記物理ブロックに移動させるブロック解放ステップを実行することを特徴とするディスクアレイにおける論理記憶領域統合方法。

【請求項3】

データを記憶するディスク装置1台以上によって構成されるドライブ列を複数と、論理記憶領域を設定する論理記憶領域設定手段と、該論理記憶領域を複数の論理ブロックに分割するブロック設定手段と、各論理ブロックに前記ドライブ列の物理ブロックを割り当てる際に前記論理記憶領域において互いに隣接する前記論理ブロックにはそれぞれ別の前記ドライブ列の物理ブロックを割り当てる割当手段と、前記複数のドライブ列を並列に動作させるドライブ制御手段とを備えるディスクアレイ装置において、

増設分のドライブ列の構成が既設のドライブ列の構成と同じである前記ドライブ列が増設されると、前記既設分のドライブ列に対応する前記論理記憶領域を第1の論理記憶領域とし、前記論理領域設定手段は該増設分のドライブ列に応じた第2の論理記憶領域と該第2の論理記憶領域とは別の第3の論理記憶領域を設定し、

前記ブロック設定手段は該第2の論理記憶領域を複数の論理ブロックに分割して、論理ブロック番号を前記第1の論理記憶領域と前記第2の論理記憶領域とで異なるように割り振る構成であり、

該第1および第2の論理記憶領域の未使用の論理ブロックをそれぞれが属していた論理記憶領域から分離させ、該分離させた論理ブロックを前記第3の論理記憶領域に属する論理ブロックとする領域併合手段と、

前記第1および第2の論理記憶領域の論理ブロックに割り当てられている前記物理ブロックのデータを前記第3の論理記憶領域の論理ブロックに割り当てられた前記物理ブロックに移動させて該論理ブロックを前記未使用の論理ブロックとするデータ移動手段と、

前記第1および第2の論理記憶領域に属していた前記論理ブロックがすべて前記第3の論理記憶領域に属するまで前記領域併合手段とデータ移動手段とを作動させ、前記第1および第2の論理記憶領域に属していた前記論理ブロックが前記第3の論理記憶領域に組み込まれる毎に前記第1および第2の論理記憶領域の論理ブロック数を前記分離させたブロック数分減算し、前記第3の論理記憶領域の論理ブロック数を該組み込んだブロック数分加算する併合制御手段とを設けた

ことを特徴とするディスクアレイ装置。

【請求項4】

請求項3記載のディスクアレイ装置において、前記第1の論理記憶領域に未使用の論理ブロックが存在しないときには、該第1の論理記憶領域のいずれかの論理ブロックに割り当てられている前記物理ブロックのデータを前記第2の論理記憶領域の未使用の論理ブロックに割り当てられている前記物理ブロックに移動させるブロック解放手段を設けたことを特徴とするディスクアレイ装置。

【請求項5】

10

20

30

40

50

請求項3または4記載のディスクアレイ装置において、

該ディスクアレイ装置の稼働中における前記ドライブ列の増設を検出する増設検知手段を設けたことを特徴とするディスクアレイ装置。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、ディスクアレイ装置に関する。

【0002】

【従来の技術】

従来、データを記憶するディスク装置1台以上によって構成されるドライブ列を複数と、論理記憶領域を設定する論理記憶領域設定手段と、論理記憶領域を複数の論理ブロックに分割するブロック設定手段と、各論理ブロックにドライブ列の物理ブロックを割り当てる際に、論理記憶領域において互いに隣接する論理ブロックにはそれぞれ別のドライブ列の物理ブロックを割り当てる割り当て手段と、複数のドライブ列を並列に動作させるドライブ制御手段とを備えるディスクアレイ装置が知られている。

10

【0003】

論理記憶領域において互いに隣接する論理ブロックにはそれぞれ別のドライブ列の物理ブロックを割り当てて、複数のドライブ列を並列に動作させることにより、外部からのアクセスに対する応答の高速化を図る手法はストライピングと呼ばれており、RAID (Redundant Arrays of Inexpensive Disks) レベル0, 3, 4, 5等に応用されている。

20

【0004】

このストライピングの一例を図8を参照して具体的に説明する。これはRAIDレベル4に相当するもので、論理ディスクにはチャンク0、1、2、3・・・が設定されていて、A～Dの4台のディスク装置に順次チャンク0, 1, 2, 3を書き込み、それらチャンク0～3に対応するパリティデータP(0-3)を残りの1台のEディスク装置に書き込む。同様に、A～Dディスク装置に順次チャンク4, 5, 6, 7を書き込み、それらデータ4～7に対応するパリティデータP(4-7)を残りのEディスク装置に書き込む。チャンク8以降についても同様である。

【0005】

このストライピングを実行すると、ファイルを書き込む際、ファイルを構成するファイルブロックが論理ディスク上の連続したチャンクに順次配置されるときには、その連続するチャンクはそれぞれ別のディスク装置のチャンクに配置される。そのため、ファイルを一貫して読み出す場合には、同一のディスク装置から連続して読み出されることがなく、あるディスク装置のチャンクからのデータ転送中に別のディスク装置のチャンクをシークしたり、さらにはキャッシュへの先読みなどを行える。このように複数のディスク装置を並列に動作させることができるので、全体として非常に高速なデータ読み出しが可能であった。また、多数のディスク装置を使用できることから大容量のデータを格納することも可能であった。

30

【0006】

つまり、大容量と高速読み出しが両立するわけで、こうした特質を活かして例えばビデオサーバ等に利用されていた。

40

【0007】

【発明が解決しようとする課題】

しかしながら、例えばビデオサーバの場合、映像データをディスクアレイ装置に蓄積していけば、それ以上データを蓄積できない満杯状態がいつかは訪れる。このとき、それ以上の映像データを記憶させようとするれば、通常のディスクアレイ装置では映像データの送出を中断し、既に蓄積済みの映像データを一旦外部の記憶装置に保存するか、または新規の大容量のディスクアレイ装置へ転送するかしか手段がなかった。何れにしてもその為必要とする時間は短くなく、その間そのビデオサーバはサービスを停止せざるを得ないのが実

50

状であった。

【0008】

また、ストライピングを行うシステムにおいてその高速性を維持するためには、ディスクアレイ装置を構成する全ディスク装置を共通の論理記憶領域下にコントロールする必要がある。ところが、単にディスク装置を増設しただけでは既設分の論理記憶領域とは別に増設分の論理記憶領域が設定されてしまうので、ストライピングの利点を活かすことができなかつた。

【0009】

このディスクアレイ装置の記憶容量を増加するためにはシステムを一旦停止しなければならないという問題、単にディスク装置を増設しただけではストライピングの利点が活かせないという問題は、ビデオサーバ以外でも存在した。本発明は、上述した問題点を解決するためになされたものであり、ストライピングを行うディスクアレイ装置において、ストライピングの利点を損なわずに、しかもその稼働を中断せずに記憶データの増加やディスク装置の増設を可能とすることを目的としている。

【0010】

【課題を解決するための手段】

上記課題を解決するための手段として、請求項1記載のディスクアレイにおける論理記憶領域統合方法は、データを記憶するディスク装置1台以上によって構成されるドライブ列を複数と、論理記憶領域を設定する論理記憶領域設定手段と、該論理記憶領域を複数の論理ブロックに分割するブロック設定手段と、各論理ブロックに前記ドライブ列の物理ブロックを割り当てる際に前記論理記憶領域において互いに隣接する前記論理ブロックにはそれぞれ別の前記ドライブ列の物理ブロックを割り当てる割当手段と、前記複数のドライブ列を並列に動作させるドライブ制御手段とを備えるディスクアレイにおいて、ドライブ列が増設された際に、増設分のドライブ列の構成が既設のドライブ列の構成と同じであることを前提として、既設のドライブ列に対応する論理記憶領域と増設分のドライブ列に対応する論理記憶領域とを統合するための論理記憶領域統合方法であって、

前記ドライブ列が増設されると、前記既設分のドライブ列に対応する前記論理記憶領域を第1の論理記憶領域とし、該増設分のドライブ列に応じた第2の論理記憶領域と該第2の論理記憶領域とは別の第3の論理記憶領域を設定し、該第2の論理記憶領域を複数の論理ブロックに分割して、論理ブロック番号を前記第1の論理記憶領域と前記第2の論理記憶領域とで異なるように割り振ってから、

該第1および第2の論理記憶領域の未使用の論理ブロックをそれぞれが属していた論理記憶領域から分離させ、該分離させた論理ブロックを前記第3の論理記憶領域に属する論理ブロックとする併合ステップと、

前記第1および第2の論理記憶領域の論理ブロックに割り当てられている前記物理ブロックのデータを前記第3の論理記憶領域の論理ブロックに割り当てられた前記物理ブロックに移動させて該論理ブロックを前記未使用の論理ブロックとするデータ移動ステップとを、

前記第1および第2の論理記憶領域に属していた前記論理ブロックがすべて前記第3の論理記憶領域に属するまで繰り返し実行し、前記第1および第2の論理記憶領域に属していた前記論理ブロックを前記第3の論理記憶領域に併合する毎に前記第1および第2の論理記憶領域の論理ブロック数を前記分離させたブロック数分減算し、前記第3の論理記憶領域の論理ブロック数を該併合されたブロック数分加算することを特徴とする。

【0011】

請求項2記載のディスクアレイにおける論理記憶領域統合方法は、請求項1記載のディスクアレイにおける論理記憶領域統合方法において、

前記併合ステップを実行する前に前記第1の論理記憶領域に未使用の論理ブロックが存在しないときには、該第1の論理記憶領域のいずれかの論理ブロックに割り当てられている前記物理ブロックのデータを前記第2の論理記憶領域の未使用の論理ブロックに割り当てられている前記物理ブロックに移動させるブロック解放ステップを実行することを特徴

10

20

30

40

50

とする。

【0012】

また、請求項1記載の方法を実行可能な装置としての請求項3記載のディスクアレイ装置は、データを記憶するディスク装置1台以上によって構成されるドライブ列を複数と、論理記憶領域を設定する論理記憶領域設定手段と、該論理記憶領域を複数の論理ブロックに分割するブロック設定手段と、各論理ブロックに前記ドライブ列の物理ブロックを割り当てる際に前記論理記憶領域において互いに隣接する前記論理ブロックにはそれぞれ別の前記ドライブ列の物理ブロックを割り当てる割り当て手段と、前記複数のドライブ列を並列に動作させるドライブ制御手段とを備えるディスクアレイ装置において、

増設分のドライブ列の構成が既設のドライブ列の構成と同じである前記ドライブ列が増設されると、前記既設分のドライブ列に対応する前記論理記憶領域を第1の論理記憶領域とし、前記論理領域設定手段は該増設分のドライブ列に応じた第2の論理記憶領域と該第2の論理記憶領域とは別の第3の論理記憶領域を設定し、

前記ブロック設定手段は該第2の論理記憶領域を複数の論理ブロックに分割して、論理ブロック番号を前記第1の論理記憶領域と前記第2の論理記憶領域とで異なるように割り振る構成であり、

該第1および第2の論理記憶領域の未使用の論理ブロックをそれぞれが属していた論理記憶領域から分離させ、該分離させた論理ブロックを前記第3の論理記憶領域に属する論理ブロックとする領域併合手段と、

前記第1および第2の論理記憶領域の論理ブロックに割り当てられている前記物理ブロックのデータを前記第3の論理記憶領域の論理ブロックに割り当てられた前記物理ブロックに移動させて該論理ブロックを前記未使用の論理ブロックとするデータ移動手段と、

前記第1および第2の論理記憶領域に属していた前記論理ブロックがすべて前記第3の論理記憶領域に属するまで前記領域併合手段とデータ移動手段とを作動させ、前記第1および第2の論理記憶領域に属していた前記論理ブロックが前記第3の論理記憶領域に組み込まれる毎に前記第1および第2の論理記憶領域の論理ブロック数を前記分離させたブロック数分減算し、前記第3の論理記憶領域の論理ブロック数を該組み込んだブロック数分加算する併合制御手段とを設けた。

【0013】

請求項2記載の方法を実行可能な装置としての請求項4記載のディスクアレイ装置は、請求項3記載のディスクアレイ装置において、前記第1の論理記憶領域に未使用の論理ブロックが存在しないときには、該第1の論理記憶領域のいずれかの論理ブロックに割り当てられている前記物理ブロックのデータを前記第2の論理記憶領域の未使用の論理ブロックに割り当てられている前記物理ブロックに移動させるブロック解放手段を設けたことを特徴とする。

【0014】

請求項5記載のディスクアレイ装置は、請求項3または4記載のディスクアレイ装置において、該ディスクアレイ装置の稼働中における前記ドライブ列の増設を検出する増設検知手段を設けたことを特徴とする。

【0015】

【発明の実施の形態】

上記の構成になる請求項1記載のディスクアレイにおける論理記憶領域統合方法は、データを記憶するディスク装置1台以上によって構成されるドライブ列を複数、論理記憶領域を設定する論理記憶領域設定手段、論理記憶領域を複数の論理ブロックに分割するブロック設定手段、各論理ブロックにドライブ列の物理ブロックを割り当てる際に論理記憶領域において互いに隣接する論理ブロックにはそれぞれ別のドライブ列の物理ブロックを割り当てる割り当て手段および複数のドライブ列を並列に動作させるドライブ制御手段を備えるディスクアレイに適用される。なお、ドライブ列は、データを記憶するディスク装置1台以上によって構成されるから、ディスク装置1台で1つのドライブ列を構成することもあ

10

20

30

40

50

## 【0016】

この論理記憶領域統合方法は、増設分のドライブ列の構成が既設のドライブ列の構成と同じであることを前提としており、ドライブ列が増設されると、既設分のドライブ列に対応する論理記憶領域を第1の論理記憶領域とし、増設分のドライブ列に応じた第2の論理記憶領域と第2の論理記憶領域とは別の第3の論理記憶領域を設定し、第2の論理記憶領域を複数の論理ブロックに分割して、論理ブロック番号を第1の論理記憶領域と第2の論理記憶領域とで異なるように割り振る。

その後、第1および第2の論理記憶領域の未使用の論理ブロックをそれぞれが属していた論理記憶領域から分離させ、分離させた論理ブロックを第3の論理記憶領域に属する論理ブロックとする併合ステップと、第1および第2の論理記憶領域の論理ブロックに割り当てられている物理ブロックのデータを第3の論理記憶領域の論理ブロックに割り当てられた物理ブロックに移動させて該論理ブロックを前記未使用の論理ブロックとするデータ移動ステップとを、第1および第2の論理記憶領域に属していた論理ブロックがすべて第3の論理記憶領域に属するまで繰り返し実行する。

10

また、第1および第2の論理記憶領域に属していた論理ブロックを第3の論理記憶領域に併合する毎に第1および第2の論理記憶領域の論理ブロック数を分離させたブロック数分減算し、第3の論理記憶領域の論理ブロック数を併合されたブロック数分加算する。

## 【0017】

つまり、第1および第2の論理記憶領域の未使用の部分を第3の論理記憶領域に併合し、その併合された部分に対応する物理ブロックに他の物理ブロックのデータを移動させることによって第1および第2の論理記憶領域に未使用の論理ブロックを作り出し、その未使用の論理ブロックを第3の論理記憶領域に併合することを繰り返す。そして、最終的には、第1および第2の論理記憶領域は消滅して第3の論理記憶領域だけになる。

20

## 【0018】

この残った第3の論理記憶領域は、既設のディスク装置および増設されたディスク装置のすべてを共通に管理する論理記憶領域となるから、ディスク装置の増設によってストライピングの利点が損なわれることはない。また、既設のディスク装置に蓄積されていたデータを他のディスクアレイに移転させるわけではないから、このディスクアレイからの読み出しは可能であり、稼働を中断させる必要はない。

## 【0019】

ところで、ディスク装置を増設した際には、例えば既設のディスク装置が満杯あるいは満杯に近くて、第1の論理記憶領域のすべての論理ブロックが使用されている可能性もある。このような場合には、いきなり併合ステップを実行するわけにはいかない。

30

## 【0020】

そこで、請求項2記載のように、併合ステップを実行する前に第1の論理記憶領域に未使用の論理ブロックが存在しないときには、第1の論理記憶領域のいずれかの論理ブロックに割り当てられている物理ブロックのデータを第2の論理記憶領域の未使用の論理ブロックに割り当てられている物理ブロックに移動させるブロック解放ステップを実行すれば、第1の論理記憶領域に未使用の論理ブロックができるから、そこで併合ステップを実行できる。なお、増設分のディスク装置は、増設時には使用されている物理ブロックがないのが普通であるから、ブロック解放ステップの実行に支障はない。

40

## 【0021】

次に、請求項1記載の方法を実行可能な装置としての請求項3記載のディスクアレイ装置においては、ドライブ列は、データを記憶するディスク装置1台以上によって構成され、ディスク装置1台で1つのドライブ列を構成することもある。このディスクアレイ装置は、そうしたドライブ列を複数備えている。

なお、増設されるドライブ列の構成は既設のドライブ列の構成と同じにされる。

## 【0022】

論理記憶領域設定手段は、ディスクアレイ装置を構成するディスク装置の記憶容量に応じて論理記憶領域を設定し、ブロック設定手段は、例えばディスクアレイ装置の構築に際

50

して設定される論理ブロックのサイズに基づいて、論理記憶領域を複数の論理ブロックに分割する。そして、割当手段は、各論理ブロックにドライブ列の物理ブロックを割り当てる際に、論理記憶領域において互いに隣接する論理ブロックにはそれぞれ別のドライブ列の物理ブロックを割り当てる。

【0023】

これによりストライピングが可能となり、ファイルを書き込む際に、ファイルを構成するファイルブロックが論理記憶領域の連続した論理ブロックに順次配置されるときには、その連続する論理ブロックはそれぞれ別のドライブ列の物理ブロックに配置される。そのため、ファイルをシーケンシャルに読み出す場合には、同一のドライブ列から連続して読み出されることはない。そこで、ドライブ制御手段により複数のドライブ列を並列に動作させれば、あるドライブ列の物理ブロックからのデータ転送中に別のドライブ列の物理ブロックをシークしたり、さらにはキャッシュへの先読みなどを行える。このように複数のドライブ列を並列に動作させることができるので、全体として非常に高速なデータ読み出しが可能である。

10

【0024】

さらに、このディスクアレイ装置においては、ドライブ列が増設されると、論理領域設定手段は、既設分のドライブ列に対応する論理記憶領域を第1の論理記憶領域として、増設分のドライブ列に応じた第2の論理記憶領域とこの第2の論理記憶領域とは別の第3の論理記憶領域を設定し、ブロック設定手段は第2の論理記憶領域を複数の論理ブロックに分割して、論理ブロック番号を第1の論理記憶領域と第2の論理記憶領域とで異なるように割り振る。

20

【0025】

領域併合手段は、第1および第2の論理記憶領域の未使用の論理ブロックをそれぞれが属していた論理記憶領域から分離させ、分離させた論理ブロックを第3の論理記憶領域に属する論理ブロックとする。

【0026】

データ移動手段は、第1および第2の論理記憶領域の論理ブロックに割り当てられている物理ブロックのデータを第3の論理記憶領域の論理ブロックに割り当てられた物理ブロックに移動させて、その論理ブロックを未使用の論理ブロックとする。併合制御手段は、第1および第2の論理記憶領域に属していた論理ブロックがすべて第3の論理記憶領域に属するまで領域併合手段とデータ移動手段とを作動させる。また、第1および第2の論理記憶領域に属していた論理ブロックを第3の論理記憶領域に併合する毎に第1および第2の論理記憶領域の論理ブロック数を分離させたブロック数分減算し、第3の論理記憶領域の論理ブロック数を併合されたブロック数分加算する。

30

【0027】

このような、各手段の作用により、請求項1について述べたとおりの効果が発揮される。また、請求項4の構成とすれば、ブロック解放手段は、第1の論理記憶領域に未使用の論理ブロックが存在しないときには、第1の論理記憶領域のいずれかの論理ブロックに割り当てられている物理ブロックのデータを第2の論理記憶領域の未使用の論理ブロックに割り当てられている物理ブロックに移動させるから、請求項2について述べたとおりの効果が得られる。

40

【0028】

さらに、請求項5記載の構成とすれば、増設検知手段によって、ディスクアレイ装置の稼働中におけるドライブ列の増設を検出することができるから、外部から特別な操作をしたり、あるいはディスク装置の増設をディスクアレイ装置に認識させるためにシステムをリセットする必要もない。

【0029】

【具体例】

次に、本発明の具体例を説明することにより発明の実施の形態を詳しく説明する。図1は、本具体例のディスクアレイ装置6の概略構成を説明するブロック図である。この図1

50

に示すように、ディスクアレイ装置 6 は、ディスクアレイ装置 6 全体の制御手段となるアレイコントローラ 8 と複数のディスク装置 5 0 とを備えている。

【 0 0 3 0 】

このアレイコントローラ 8 は、論理記憶領域設定手段、ブロック設定手段、割当手段、領域併合手段、データ移動手段、併合制御手段、ブロック解放手段および増設検知手段として機能する CPU 3 0 を中心にして構成されている。CPU 3 0 には、CPU 3 0 が実行するプログラムを格納している ROM 2 0、CPU 3 0 のワークエリアとなる RAM 1 0、図示しないホスト装置とのデータの送受などを行うための通信手段としてのインタフェース ( I / F ) 3 5、ドライブ制御手段であり CPU 3 0 と共同してデータ移動手段およびブロック解放手段として機能する複数のドライブ制御装置 4 0 が接続されている。

10

【 0 0 3 1 】

各ドライブ制御装置 4 0 には、それぞれ複数のディスク装置 5 0 が、バスライン 5 5 によりパラレルに接続されている。図においては、各ドライブ制御装置 4 0 は、すでに 3 台のディスク装置 5 0 ( ドライブ 0、1、2 ) が接続されていて、そこに新たに 3 台のディスク装置 5 0 ( ドライブ 3、4、5 ) が増設された状態を示している。各ドライブ制御装置 4 0 の同じドライブ番号 ( 例えばドライブ 0 ) 同士によって 1 つのドライブ列が構成されており、ディスク装置 5 0 の増設前にあってはドライブ列 0、1、2 の 3 つのドライブ列が存在し、増設後はドライブ列 3、4、5 が増えて 6 つのドライブ列が存在している。なお、ディスク装置 5 0 は、いわゆる物理的なディスクドライブユニットとそれを制御するコントロールユニットとが一体化されたものである。

20

【 0 0 3 2 】

アレイコントローラ 8 の CPU 3 0 は、ROM 2 0 内のアレイ管理ソフトウェア等に基づき、各ドライブ制御装置 4 0 を介して各ドライブ列のディスク装置 5 0 を同期させて動作させることができ、しかも複数のドライブ列を並列に動作させることができる。

【 0 0 3 3 】

本具体例では、CPU 3 0 が、システムの起動時に接続されているディスク装置 5 0 の台数と記憶容量とを認識して、総記憶容量に応じた論理記憶領域を設定し、図 2 に示される論理ブロック 物理ブロック変換テーブルを作成し RAM 1 0 に記憶させておき、このテーブルに基づいてディスク装置 5 0 へのデータの書き込み、読み出しを行う。

【 0 0 3 4 】

ホスト装置から例えば論理ブロック番号 0 ~ 2 のデータの読み出し要求を受けたとすると、CPU 3 0 は、まず論理ブロック番号 0 に対応するドライブ列 ( ドライブ列番号 0 ) の物理ブロック番号 0 の読み出しを、各ドライブ制御装置 4 0 に指示する。すると、各ドライブ制御装置 4 0 が、この指示に応じてドライブ 0 のディスク装置 5 0 を作動させて、それぞれの物理ブロック 0 に記録されているデータを読み出させる。読み出されたデータは、インタフェース 3 5 を通じてホスト装置に送出されるのであるが、CPU 3 0 は、ドライブ列 0 からの読み出しと並行させて、ドライブ列 1 のディスク装置 5 0 を作動させて物理ブロック 0 のシークや先読みを行わせ、あるいはさらにドライブ列 2 の物理ブロック 0 のシークや先読みを行わせる。このように、例えばドライブ列 0 から連続して読み出しすることをせずに、ドライブ列 0 ~ 2 を並列に作動させることで、ホスト装置からのアクセス要求に高速で対応することができる。

30

40

【 0 0 3 5 】

ところが、図 1 に示されるようにドライブ列 3 ~ 5 が増設された際に、ドライブ列 0 ~ 2 に対応する論理記憶領域とドライブ列 3 ~ 5 に対応する論理記憶領域が別々であると、すなわち図 2 に示されるようなブロック 物理ブロック変換テーブルを 2 つ持っていたのでは、ドライブ列 0 ~ 2 におけるストライピングとドライブ列 3 ~ 5 におけるストライピングが別々になってしまうから、ディスクアレイ装置 6 全体としての記憶容量は増加するとしても、高速性の点では不十分と言わざるを得ない。

【 0 0 3 6 】

そこで、本具体例のディスクアレイ装置 6 では、次に説明するように既設のドライブ列

50



0 ~ 2 と増設されたドライブ列 3 ~ 5 の論理記憶領域を統合している。ディスクアレイ装置 6 が稼働しているときには、アレイコントローラ 8 の CPU 30 は、図 3 に示されるディスクアレイメイン処理を実行している。この処理においては、CPU 30 は、ホスト装置からのアクセス要求があるか否かを判断し（ステップ 100、以下ステップを S と略記する）、アクセス要求があれば（S100：YES）、その要求に応じたディスクアレイアクセス処理（書き込みまたは読み出し）を実行する（S110）。ホスト装置からのアクセス要求がなかった場合（S100：NO）または S110 の実行後に、CPU 30 は、各ドライブ制御装置 40 に接続されているディスク装置 50 の台数を取得して、ドライブ列の増設があったか否かを判断する（S120）。もし、ドライブ列の増設があれば（S120：YES）、CPU 30 は領域併合処理を起動して S100 に戻る。なお、CPU 30 は、この領域併合処理をディスクアレイメイン処理と並列に実行することができる。また、ドライブ列の増設がなければ（S120：NO）、そのまま S100 に戻る。

10

**【0037】**

図 4 に示すように、CPU 30 は、領域併合処理を開始すると、まず増設されたドライブ列（図 1 に示される例ではドライブ列 3 ~ 5）の合計の記憶容量に対応する新たな論理記憶領域（新領域）を設定する（S200）。この新領域の設定では、新領域の先頭論理ブロックアドレスを併合後の論理ブロックアドレスの末尾 + 1 とし、論理ブロック数を増設される論理ブロックの数として、論理ブロック 物理ブロック変換テーブルが作成される。なお、論理ブロックのサイズは予め設定されているから、増設されたドライブ列 3 ~ 5 の記憶容量が判れば、増設される論理ブロックの数を決定できる。また、この増設される論理ブロック数と既設のドライブ列 0 ~ 2 の論理ブロック数の和が併合後の論理ブロック数になる。CPU 30 は、このようにして新領域の「論理ブロック 物理ブロック変換テーブル」が作成されると、既設分のドライブ列 0 ~ 2 の論理記憶領域（旧領域）の「論理ブロック 物理ブロック変換テーブル」と合わせて、図 5（a）に示される構造の論理ブロック 物理ブロック変換テーブルとして RAM 10 に記憶させる。図 5（a）では、論理ブロック番号 0 ~ 8 が既設分（旧領域）、論理ブロック番号 18 ~ 26 が増設分（新領域）である。

20

**【0038】**

次に、CPU 30 は、旧領域と新領域とを併合するための論理記憶領域（併合領域）を設定する（S210）。設定時の併合領域の先頭論理ブロックアドレスは新領域の先頭論理ブロックアドレス（図 5（a）の例では論理ブロック番号 18）であり、論理ブロック数は 0 である。この併合領域が設定された際の状態を、図 6（a）、（イ）に示す。

30

**【0039】**

続いて、CPU 30 は、この領域併合処理が終了するまで（併合完了まで）、ディスクアレイ装置 6 に新たなファイルの書き込みを禁じるための処理を行う（S220）。後述するように、この領域併合処理においては未使用の論理ブロック（当然ながらその論理ブロックに割り当てられている物理ブロックも未使用）の存在が不可欠である。したがって、S220 の処理（書き込み禁止）により未使用の論理ブロックが無くなってしまふことを防止している。

**【0040】**

次に、CPU 30 は、旧領域の末尾部分に有効データが存在するか否かを判断する（S230）。図 5（a）に示される状態では論理ブロック番号 8 が末尾となっている。なお、この末尾部分は、以下の処理が進むに従って変更される。また、末尾部分という場合、必ずしも末尾の 1 ブロックだけを指しているわけではない。

40

**【0041】**

旧領域の末尾部分に有効データがあれば（S230：YES）、併合領域に未使用の論理ブロックがあるか否かを判断する（S240）。そして、併合領域に未使用の論理ブロックがあれば（S240：YES）、旧領域の末尾部分のデータを併合領域の未使用の論理ブロックに移動する。具体的には、CPU 30 は、各ドライブ制御装置 40 に指示して、旧領域の末尾部分の論理ブロックに割り当てられている物理ブロックのデータを併合領

50

域の未使用の論理ブロックに割り当てられている物理ブロックに移動させる。

【 0 0 4 2 】

また、併合領域に未使用の論理ブロックがなければ ( S 2 4 0 : N O )、C P U 3 0 は、旧領域の末尾部分のデータを新領域の先頭方向にある空きの論理ブロックへ移動する ( S 2 6 0 )。具体的には、C P U 3 0 は、各ドライブ制御装置 4 0 に指示して、旧領域の末尾部分の論理ブロックに割り当てられている物理ブロックのデータを新領域の先頭方向にある未使用の論理ブロックに割り当てられている物理ブロックに移動させる。ここで、先頭方向とは、論理ブロックアドレスが小さい側ということであり、空きの論理ブロックが複数あればその中で論理ブロックアドレスが最小のものという意味である。

【 0 0 4 3 】

なお、この領域併合処理が起動された直後 ( ドライブ列が増設された直後 ) では、併合領域は「空」 ( S 2 1 0 参照 ) で、未使用の論理ブロックもないから、S 2 4 0 で否定判断となり、S 2 6 0 が実行されることになる。S 2 5 0 または S 2 6 0 の処理後または S 2 3 0 で否定判断の場合には、C P U 3 0 は、新領域の末尾部分に有効データが存在するか否かを判断する ( S 2 7 0 )。新領域の末尾部分に有効データが存在すれば ( S 2 7 0 : Y E S )、C P U 3 0 は、その有効データを併合領域へ移動してから ( S 2 8 0 )、S 2 9 0 に進む。また、新領域の末尾部分に有効データが存在しなければ ( S 2 7 0 : N O )、C P U 3 0 は、S 2 8 0 を実行することなく、S 2 9 0 に進む。なお、この領域併合処理が起動された直後 ( ドライブ列が増設された直後 ) では、増設分のドライブ列には有効データは存在しないはずであるから、S 2 7 0 で否定判断となり、S 2 8 0 は実行されない。

【 0 0 4 4 】

S 2 3 0 ~ S 2 6 0 の処理により旧領域の末尾部分の論理ブロック ( 正確にはこの論理ブロックに割り当てられている物理ブロック ) には有効データは存在しないからこの論理ブロックを旧領域から切り離しても問題はない。そこで、S 2 9 0 では、C P U 3 0 は、旧領域の末尾部分の論理ブロックを切り離し、旧領域のブロック数を切り離れたブロック数分減算する。

【 0 0 4 5 】

続いて、C P U 3 0 は、新領域の末尾部分を切り離し、新領域の論理ブロック数を切り離れたブロック数分減算する ( S 3 0 0 )。S 2 7 0 ~ S 2 8 0 の処理により新領域の末尾部分の論理ブロック ( 正確にはこの論理ブロックに割り当てられている物理ブロック ) には有効データは存在しないからこの論理ブロックを新領域から切り離しても問題はない。

【 0 0 4 6 】

次に C P U 3 0 は、S 2 9 0 および S 3 0 0 において切り離された新旧領域の末尾部分を併合領域の先頭側に結合して、併合領域の先頭論理ブロックアドレスを結合したブロック数分減算し、併合領域の論理ブロック数を結合したブロック数分加算して、「論理ブロック 物理ブロック変換テーブル」を更新する ( S 3 1 0 )。S 2 9 0 で旧領域の論理ブロック番号 8 番であった論理ブロックが切り離されて、S 3 0 0 では新領域の論理ブロック番号 2 6 番であった論理ブロックが切り離されたとして、この S 3 1 0 においては、論理ブロックが 2 つ結合されるから、併合領域の先頭論理ブロックアドレスは、1 8 ( S 2 1 0 で設定 ) - 2 = 1 6、併合領域の論理ブロック数は 0 ( S 2 1 0 で設定 ) + 2 = 2 となる。

【 0 0 4 7 】

次に、C P U 3 0 は、例えば併合領域の末尾論理ブロックアドレスに基づいて、新旧領域にあった全ての論理ブロックが併合領域に結合されたか否かを判断し ( S 3 2 0 )、否定判断であれば S 2 3 0 に回帰し、肯定判断であればこの領域併合処理を終了する。

【 0 0 4 8 】

S 2 3 0 ~ S 3 1 0 の処理を何回か実行すると、「論理ブロック 物理ブロック変換テーブル」は図 5 ( b ) に例示されるようになり、旧領域、新領域および併合領域の状態は

10

20

30

40

50

、図6(b)、(ロ)に示すようになっていて、さらにS230~S310の処理が繰り返されて併合完了直前となると「論理ブロック 物理ブロック変換テーブル」は図5(c)に例示されるようになり、旧領域、新領域および併合領域の状態は、図6(c)、(ハ)に示すようになっていて、併合が完了した際には、「論理ブロック 物理ブロック変換テーブル」は図5(d)に例示されるようになり、図6(d)、(ニ)に示すように、旧領域および新領域は消滅し(厳密には空領域となり)併合領域のみが存在している。

#### 【0049】

以上のように、本具体例のディスクアレイ装置6では、ドライブ列の増設があると、CPU30がこれを認識して領域併合処理を起動する。この領域併合処理においては、旧領域の末尾部分に有効データがなければ、そのまま末尾部分を切り離し、旧領域の末尾部分に有効データがあれば、そのデータを併合領域または新領域に移動させることによって、この末尾部分から有効データを無くしてから、その末尾部分を切り離す。同様に、新領域の末尾部分に有効データがなければ、そのまま末尾部分を切り離し、新領域の末尾部分に有効データがあれば、そのデータを併合領域に移動させることによって、この末尾部分から有効データを無くしてから、その末尾部分を切り離す。そして、切り離された新旧領域の末尾部分を併合領域の先頭側に結合する。こうした処理を繰り返すことによって、新旧2つの論理記憶領域を併合している。この併合後の論理記憶領域は、既設のドライブ列0~2および増設されたドライブ列3~5のすべてを共通に管理する論理記憶領域となるから、ドライブ列の増設によってストライピングの利点が損なわれることはない。また、既設のドライブ列に蓄積されていたデータを他のディスクアレイ装置に移転させるわけではないから、このディスクアレイ装置6からの読み出しは可能であり、稼働を中断させる必要はない。

(変形例)

上述の例はディスクアレイ装置6の稼働中にドライブ列の増設を認識して、領域併合処理を起動する例であるが、この処理をシステムの起動時(正確には再起動時)に実行させる構成も可能である。図7にそのための処理の一例を示す。

#### 【0050】

図7に示すように、ディスクアレイ装置6が起動された直後のディスクアレイ初期化処理において、CPU30は、ドライブ列の増設があったか否かを判断し(S400)、ドライブ列の増設があれば(S400:YES)、領域併合処理を起動する(S410)。

#### 【0051】

このような構成とすれば、例えば定期点検や何らかのトラブルでディスクアレイ装置6を停止させた際などにドライブ列を増設したときには、再起動した直後にその増設に対処できる。また、具体例の構成(稼働中にドライブ列の増設を認識して領域併合処理を起動する構成)と、この変形例の構成とを併せ持つ構成とすれば、ドライブ列の増設に一層適切に対処できる。

#### 【0052】

以上、具体例に従って、本発明の実施の形態について説明したが、本発明はこのような具体例に限定されるものではなく、本発明の要旨を逸脱しない範囲でさまざまに実施できることは言うまでもない。例えば、具体例ではディスクアレイメイン処理でドライブ列の増設の有無を判断しているが、ドライブ列の増設の有無を判断して、増設があれば領域併合処理を起動するウォッチ処理を一定の周期、例えば1時間周期で実行する構成や割り込み信号が有る毎にウォッチ処理を実行する構成としてもよい。

#### 【0053】

#### 【発明の効果】

以上説明したように、請求項1記載のディスクアレイにおける論理記憶領域統合方法によれば、ディスクアレイにおいてディスク装置が増設された際に、既設分に対応する論理記憶領域と増設分に対応する論理記憶領域とを併合して、統合された論理記憶領域(併合論理記憶領域)にすることができる。

10

20

30

40

50

## 【 0 0 5 4 】

この統合された論理記憶領域は、既設のディスク装置および増設されたディスク装置のすべてを共通に管理する論理記憶領域となるから、ディスク装置の増設によってストライピングの利点が損なわれることはない。また、既設のディスク装置に蓄積されていたデータを他のディスクアレイに移転させるわけではないから、このディスクアレイからの読み出しは可能であり、稼働を中断させる必要はない。

## 【 0 0 5 5 】

また請求項 2 記載の構成とすれば、既設のディスク装置が満杯あるいは満杯に近くても、請求項 1 記載の方法を実施できる。請求項 3 記載のディスクアレイ装置は、請求項 1 記載の方法を実行するのに好適である。

10

## 【 0 0 5 6 】

請求項 4 のディスクアレイ装置は、請求項 2 記載の方法を実行するのに好適である。さらに、請求項 5 記載の構成とすれば、増設検知手段によって、ディスクアレイ装置の稼働中におけるドライブ列の増設を検出することができるから、外部から特別な操作をしたり、あるいはディスク装置の増設をディスクアレイ装置に認識させるためにシステムをリセットする必要もない。

## 【 図面の簡単な説明 】

【 図 1 】 具体例のディスクアレイ装置の構成を説明するブロック図である。

【 図 2 】 具体例のディスクアレイ装置の CPU が構築する論理ブロック 物理ブロック変換テーブルの一例の説明図である。

20

【 図 3 】 具体例のディスクアレイ装置の CPU が実行するディスクアレイメイン処理のフローチャートである。

【 図 4 】 具体例のディスクアレイ装置の CPU が実行する領域併合処理のフローチャートである。

【 図 5 】 具体例のディスクアレイ装置における領域併合処理での論理ブロック 物理ブロック変換テーブルの変遷の説明図であり、図 5 ( a ) は併合開始前、図 5 ( b ) は併合開始直後、図 5 ( c ) は併合完了直前、図 5 ( d ) は併合完了直後の論理ブロック 物理ブロック変換テーブルの説明図である。

【 図 6 】 具体例のディスクアレイ装置における各領域の変遷の説明図であり、図 6 ( a ) は併合開始前、図 6 ( b ) は併合開始直後、図 6 ( c ) は併合完了直前、図 6 ( d ) は併合完了直後における各領域の関係の説明図、図 6 ( e ) は併合開始前、図 6 ( f ) は併合開始直後、図 6 ( g ) は併合完了直前、図 6 ( h ) は併合完了直後における論理ブロックアドレスの説明図である。

30

【 図 7 】 変形例のディスクアレイ初期化処理のフローチャートである。

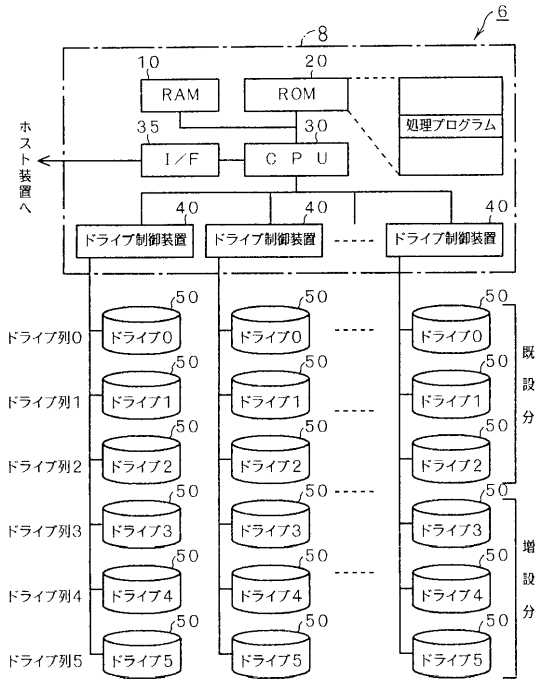
【 図 8 】 ストライピングの説明図である。

## 【 符号の説明 】

6・・・ディスクアレイ装置、8・・・アレイコントローラ、10・・・RAM、20・・・ROM、30・・・CPU ( 論理記憶領域設定手段、ブロック設定手段、割当手段、領域併合手段、データ移動手段、併合制御手段、ブロック解放手段、増設検知手段 )、35・・・インタフェース、40・・・ドライブ制御装置 ( ドライブ制御手段、データ移動手段、ブロック解放手段 )、50・・・ディスク装置、55・・・バスライン。

40

【図1】

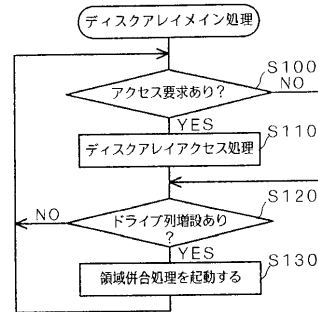


【図2】

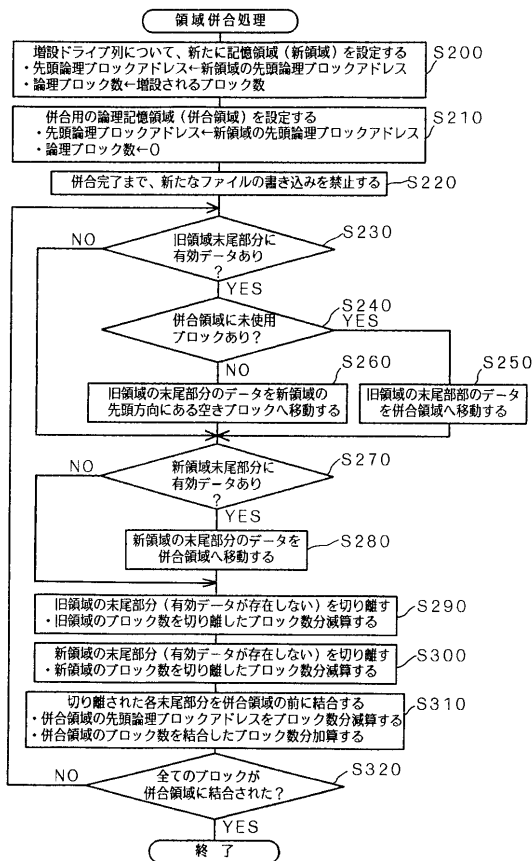
増設前 論理ブロック→物理ブロック変換テーブル

論理ブロック番号	ドライブ列番号	物理ブロック番号
0	0	0
1	1	0
2	2	0
3	0	1
4	1	1
5	2	1
6	0	2
7	1	2
8	2	2

【図3】



【図4】



【図5】

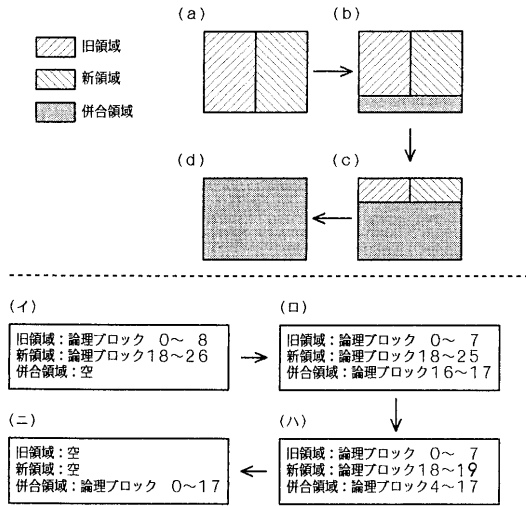
併合処理における論理ブロック→物理ブロック変換テーブル

(a) 増設後・併合開始前			(b) 増設後・併合開始直後		
論理ブロック番号	ドライブ列番号	物理ブロック番号	論理ブロック番号	ドライブ列番号	物理ブロック番号
0	0	0	0	0	0
1	1	0	1	1	0
2	2	0	2	2	0
3	0	1	3	0	1
4	1	1	4	1	1
5	2	1	5	2	1
6	0	2	12	0	2
7	1	2	13	1	2
8	2	2	14	2	2
18	3	0	15	3	2
19	4	0	16	4	2
20	5	0	17	5	2
21	3	1	18	3	0
22	4	1	19	4	0
23	5	1	20	5	0
24	3	2	21	3	1
25	4	2	22	4	1
26	5	2	23	5	1

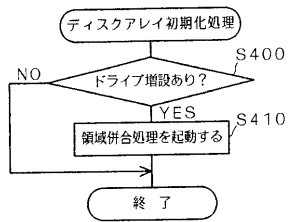
  

(c) 増設後・併合完了直前			(d) 増設後・併合完了直後		
論理ブロック番号	ドライブ列番号	物理ブロック番号	論理ブロック番号	ドライブ列番号	物理ブロック番号
0	0	0	0	0	0
1	1	0	1	1	0
2	2	0	2	2	0
6	0	1	3	3	0
7	1	1	4	4	0
8	2	1	5	5	0
9	3	1	6	0	1
10	4	1	7	1	1
11	5	1	8	2	1
12	0	2	9	3	1
13	1	2	10	4	1
14	2	2	11	5	1
15	3	2	12	0	2
16	4	2	13	1	2
17	5	2	14	2	2
18	3	0	15	3	2
19	4	0	16	4	2
20	5	0	17	5	2

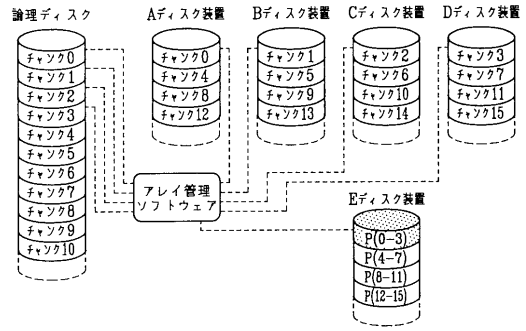
【 図 6 】



【 図 7 】



【 図 8 】



---

フロントページの続き

(58)調査した分野(Int.Cl. , D B名)

G06F 3/06- 3/08

G06F 12/00-12/16

G11B 20/00-20/18