

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2011-118623

(P2011-118623A)

(43) 公開日 平成23年6月16日(2011.6.16)

(51) Int.Cl. F 1 テーマコード (参考)
G 0 6 F 1 2 / 0 0 (2006.01) G 0 6 F 1 2 / 0 0 5 2 0 P 5 B 0 8 2

審査請求 未請求 請求項の数 1 O L (全 8 頁)

(21) 出願番号	特願2009-274876 (P2009-274876)	(71) 出願人	000001007
(22) 出願日	平成21年12月2日 (2009. 12. 2)		キヤノン株式会社
			東京都大田区下丸子3丁目30番2号
		(74) 代理人	100090273
			弁理士 國分 孝悦
		(72) 発明者	鈴木 靖教
			東京都大田区下丸子3丁目30番2号 キ
			ヤノン株式会社内
		Fターム(参考)	5B082 EA01 GA15

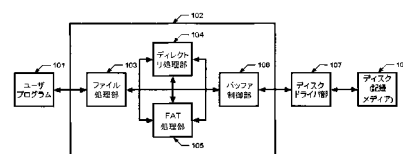
(54) 【発明の名称】 データ同期装置

(57) 【要約】

【課題】 F A T型ディスク上のファイルへのデータ書き込みや削除といった動作を行う場合に、データの更新が必要となるF A T領域へのディスク入出力処理を軽減できるようにする。

【解決手段】 F A Tのデータを変更する手段と、データの読み出しまたは書き込みを行う際の一時保存エリアの制御を行う手段と、前記一時保存エリアに記憶されたデータのうち、データの変更に応じて記録媒体へ書き込む手段とを有し、少なくとも1つのF A Tのデータを前記一時保存エリアから読み出して変更し、前記変更したデータを前記記録媒体に書き込む位置と、内容を重複して保存する他のF A Tの書き込み位置とをF A Tの大きさを用いて算出して、前記一時保存エリアに記憶するよう制御し、さらに、前記一時保存エリアに記憶された変更されたF A Tデータを記録媒体へ書き戻す際、他のF A Tの書き込み位置へ同一のデータを書き込む。

【選択図】 図1



【特許請求の範囲】

【請求項 1】

ファイルアロケーションテーブルの大きさの情報を取得する取得手段と、
前記ファイルアロケーションテーブルのデータを変更する F A T 処理手段と、
データの読み出しまたは書き込みを行う際の一時保存エリアの制御を行うバッファ制御手段と、

前記一時保存エリアに記憶されたデータのうち、データの変更に応じて記録媒体へ書き込む同期手段とを有し、

前記 F A T 処理手段は、少なくとも 1 つのファイルアロケーションテーブルのデータを前記一時保存エリアから読み出して変更し、前記変更したデータを前記記録媒体に書き込む位置と、同一の内容を保存する他のファイルアロケーションテーブルの書き込み位置とを前記取得手段により取得したファイルアロケーションテーブルの大きさを用いて算出して、前記一時保存エリアに記憶するよう前記バッファ制御手段を制御し、

前記同期手段は、前記一時保存エリアに記憶された変更されたファイルアロケーションテーブルのデータを記録媒体へ書き戻す際、前記他のファイルアロケーションテーブルの書き込み位置へ同一のデータを書き込むことを特徴とするデータ同期装置。

【発明の詳細な説明】

【技術分野】

【0001】

本発明はデータ同期装置に関し、特に、F A T ファイルシステムに用いて好適な技術に関する。

【背景技術】

【0002】

ファイルアロケーションテーブル (File Allocation Table、以下、F A T) を用いてファイルデータを管理するファイルシステムは、一般に F A T ファイルシステムと呼ばれている。特に、F A T ファイルシステム用に論理フォーマットされたハードディスク等の記録メディアは、F A T 型ディスクと呼ばれている。ユーザプログラムからの指示を受け、F A T 型ディスクに対してデータの読み出しや書き込みといった動作を行う場合、F A T ファイルシステムでは、その入出力をセクタと呼ばれる単位で行う。ここで、1 セクタの大きさは通常は 512 バイト、またはそのべき乗である。また、2 のべき乗 (1、2、4、8、16、…) 個のセクタの集合はクラスタと呼ばれており、F A T 型ディスク上に存在するファイルはクラスタの集合で表される。1 クラスタが何個のセクタから構成されるかは主に F A T 型ディスクの容量によってディスクの論理フォーマット時に決定される。

【0003】

F A T ファイルシステムにより、新規にファイルを作成する場合は以下のように行われる。まず、データ内容を記録するために、初めてファイルに未使用クラスタが割り付けられると、対応する F A T 項目の値は未使用を示す 0000H から、使用中かつ最終クラスタであることを示す FFFFH に変更される。そして、同じファイルに対してデータを追加記録し続けると、データ内容の大きさが 1 クラスタ分では足りなくなる。そこで、ファイルの最終クラスタに対応する F A T 項目の値は次に新しく追加して割り付けられる未使用クラスタの位置を示すクラスタ番号に変更される。このとき、新しく追加された未使用クラスタの F A T 項目の値を 0000H から FFFFH に変更する。これを繰り返すことにより 1 つのファイルに次々とクラスタを割り付けていくことができる。以前に作成したファイルが存在し、それにデータを追加していく場合も同様である。一方、ファイルの大きさを小さくする場合、もしくはファイルを削除する場合には、操作の結果そのファイルに割り当てておく必要の無くなったクラスタに対応する F A T 項目の値を 0000H に変更する。そして、新たなファイルの大きさに対して最終クラスタに対応する F A T 項目の値を FFFFH に変更する。

【0004】

ここで、前述したようにディスク上のデータは、セクタ単位で入出力が行われる。した

10

20

30

40

50

がって、F A Tの内容を更新する際は、ディスク上のデータを直接変更するのではなく、高速に入出力できる半導体メモリなどに一時保存エリア（以下、バッファ）を設ける。そして、一旦そのバッファにディスク上の元データを蓄積し、これに対して必要な変更を行い、その後適宜バッファの内容をディスク上へ書き戻すといった処理が行われる。このように、バッファの内容をディスク上へ書き戻す処理はデータ同期と呼ばれている（例えば、特許文献1参照）。

【先行技術文献】

【特許文献】

【0005】

【特許文献1】特許第3703181号公報

10

【発明の概要】

【発明が解決しようとする課題】

【0006】

F A T型ディスクにおいては、F A T 1及びF A T 2の内容を常に同じものにしておく必要がある。そのため、ファイルへのデータの追加書き込みや削除などの動作に伴ってF A Tを更新する際には、F A T 1及びF A T 2の内容を、それぞれ一旦バッファ上へ読み出す。そして、それぞれに対して同じ変更を行ってからディスクに書き戻してデータを同期するといった処理が必要になる。

【0007】

また、F A T型ディスクの場合、F A T領域で各ファイルへのクラスタの割り当て状況を管理するという仕組みから、1つのファイルが使用しているクラスタの情報はF A T領域に散在している可能性がある。そのため、たとえファイルのサイズが小さくとも、更新する必要のあるF A T 1及びF A T 2のセクタの数が多くなる場合がある。C P U処理と比較して非常に多くの処理時間を要するディスクへの入出力動作が多くなって、結果的にファイル操作全体の処理時間が長くなるという問題がある。

20

【0008】

本発明は前述の問題点に鑑み、F A T型ディスク上のファイルへのデータ書き込みや削除といった動作を行う場合に、データの更新が必要となるF A T領域へのディスク入出力処理を軽減できるようにすることを目的としている。

【課題を解決するための手段】

30

【0009】

本発明のデータ同期装置は、ファイルアロケーションテーブルの大きさの情報を取得する取得手段と、前記ファイルアロケーションテーブルのデータを変更するF A T処理手段と、データの読み出しまたは書き込みを行う際の一時保存エリアの制御を行うバッファ制御手段と、前記一時保存エリアに記憶されたデータのうち、データの変更に応じて記録媒体へ書き込む同期手段とを有し、前記F A T処理手段は、少なくとも1つのファイルアロケーションテーブルのデータを前記一時保存エリアから読み出して変更し、前記変更したデータを前記記録媒体に書き込む位置と、同一の内容を保存する他のファイルアロケーションテーブルの書き込み位置とを前記取得手段により取得したファイルアロケーションテーブルの大きさを用いて算出して、前記一時保存エリアに記憶するよう前記バッファ制御手段を制御し、前記同期手段は、前記一時保存エリアに記憶された変更されたファイルアロケーションテーブルのデータを記録媒体へ書き戻す際、前記他のファイルアロケーションテーブルの書き込み位置へ同一のデータを書き込むことを特徴とする。

40

【発明の効果】

【0010】

本発明によれば、重複する一方のファイルアロケーションテーブルのバッファ上への読み出し処理を省略することができる。これにより、その分高速にファイルデータの書き込みや、ファイルの削除といった動作を行うことができる。

【図面の簡単な説明】

【0011】

50

【図１】データ同期装置の機能構成例を示すブロック図である。

【図２】データ同期装置のハードウェア構成例を示すブロック図である。

【図３】F A T型ディスクのフォーマットの一例を示す図である。

【図４】F A T領域の構造例を示す図である。

【図５】ルートディレクトリ領域に記録されている情報の一例を示す図である。

【図６】バッファの構成例を示す図である。

【図７】F A Tの更新処理手順の一例を示すフローチャートである。

【図８】バッファデータの同期処理の手順の一例を示すフローチャートである。

【発明を実施するための形態】

【００１２】

10

（第１の実施形態）

以下、本発明の第１の実施形態について図面を参照しながら説明する。

図１は、本実施形態に係るファイルアロケーションテーブル（F A T）を用いたF A Tファイルシステムにおけるデータ同期装置の機能構成例を示すブロック図である。

図１において、１０１はF A Tファイルシステムを利用するユーザプログラムであり、１０２は本実施形態によるF A Tデータ同期方式を実現するF A Tファイルシステムプログラムである。１０３はユーザプログラム１０１からの要求を受け付け、ファイルディסקリプタを管理してファイル名などで指定されたファイルを開閉する処理や、新規ファイルの作成、読み出し、書き込み動作のファイルポインタの管理などを行うファイル処理部である。

20

【００１３】

１０４はディスクから読み出されたディレクトリエントリ情報を解析して、ユーザプログラム１０１から指示されたファイルの存在を検索するディレクトリ処理部である。また、ディレクトリ処理部１０４は、ファイルの先頭部分のデータの含まれるクラスタ（開始クラスタ）番号を取得するとともに、最新のファイル情報をディレクトリエントリに書き込むといったディレクトリエントリの解析や更新等を行う。１０５は読み出し及び書き込みの対象となるファイルのファイル位置からディスク上のクラスタ番号をディレクトリエントリやF A T領域の情報を解析して取得するF A T処理部である。また、F A T処理部１０５は、ファイルヘデータを追加したり、ファイルを削除した場合には当該ファイルのF A Tチェーンを正しく形成してF A T領域に書き込んだりする処理などを行う。

30

【００１４】

１０６はファイル処理部１０３、ディレクトリ処理部１０４及びF A T処理部１０５からの要求により、内部管理する一時保存エリアのバッファ内に格納されている指定されたセクタのデータを要求元へ返す処理を行うバッファ制御部である。また、バッファ制御部１０６は、指示に従って変更されたバッファの内容をディスクに書き戻す処理も行う。１０７は実際のデータの格納領域となるハードディスク、またはフラッシュメモリカードといった記録媒体のディスク（記録メディア）１０８から各種データを読み出す、またはディスク１０８へデータを書き込むディスクドライバ部である。

【００１５】

図２は、本実施形態に係るデータ同期装置のハードウェア構成例を示すブロック図である。

40

図２において、２０１はC P Uであり、上述のユーザプログラム１０１やF A Tファイルシステムプログラム１０２をロードして実行する。２０２はR A Mであり、上述のユーザプログラム１０１やF A Tファイルシステムプログラム１０２、ディスクドライバ部１０７を機能させるプログラムを格納する。さらにR A M ２０２は、バッファ制御部１０６によるディスク１０８上のデータの読み出し、または書き込みの際のバッファ領域として使用される。また、各ソフトウェアプログラムの動作時に使用する各種変数の退避、或は保存用のワークエリアとして使用される。２０３はファイルデータの格納領域となるF A T型のディスク１０８との間で読み出し及び書き込みを行うためのディスク装置である。２０４はC P U ２０１、R A M ２０２、及びディスク装置２０３を接続するシステムバス

50

である。

【0016】

図6は、バッファ制御部106により制御されるバッファの構成例を示す図である。

図6において、バッファは、バッファの状態を示す状態ビット601、そのバッファの内容が存在するセクタ番号602、及びディスク108からバッファへとコピーされるデータ603本体に区分される。状態ビット601には、そのバッファの内容がディスク108への書き出しを要することを示すdirtyビットが含まれている。なお、それ以外の状態については任意であるとする。以上の構成のバッファは、RAM202上にユーザプログラム101が動作するために十分な領域が確保される。

【0017】

次に、本実施形態におけるFATファイルシステムによるFATの更新処理動作について説明する。なお、本実施形態におけるFAT型のディスク108の記録フォーマット、ディレクトリエントリの構成、およびFAT領域の構成はそれぞれ、図3、図4及び図5に示すものである。

【0018】

図3は、FAT型ディスクのフォーマットの一例を示す図であり、基本的な構造は、MS-DOS（登録商標）で採用されているものと同じである。

図3において、301は先頭の1セクタ目（論理セクタ番号0）を示すブートセクタである。ブートセクタ301には、オペレーティングシステムをロードするためのブートプログラム、前述のセクタ／クラスタ数、後述するFAT領域に使用されているセクタの数、ルートディレクトリエントリの個数などの情報が記録されている。

【0019】

302はFAT領域であり、クラスタの所在とファイル本体（内容）へのクラスタの割り付け（アロケーション）状況とを兼ねたそのクラスタ番号に続く次のクラスタ番号を示すFAT項目を記録する領域である。FAT領域302は安全性を確保するために2重化されており、図3における第1のFAT領域（以降、FAT1）と第2のFAT領域（以降、FAT2）とは常に同一の内容のものが記録されるようになっている。また、FAT1とFAT2はディスク上に連続して配置されている。

【0020】

図4は、FAT領域302の構造例を示す図である。現在一般に広く使用されているFAT型ディスクには、1クラスタ分のFAT項目を示すために、12ビット、16ビット、または32ビットを使用するものがあるが、本実施形態では16ビットの場合を例に説明する。

図4において、各FAT項目の先頭からの位置は、そのままその項目で管理しているクラスタの位置に対応している。なお、各項目の初期値は0000H（16進数0）であり、これは未使用状態を示している。

【0021】

図3の説明に戻り、303はルートディレクトリ領域であり、ファイルの所在を管理する領域である。1つのファイルに対応する項目は32バイトで表現される。ルートディレクトリ領域303には、図5に示すように、ファイルの名前や大きさ（ファイルサイズ）、そのファイルのデータが含まれている先頭のクラスタ（開始クラスタ）番号などの情報が記録されている。

【0022】

304はデータ領域であり、実際のファイルの内容（データ本体）やルートディレクトリに属するディレクトリに関する情報を記録する領域である。FAT領域302と同様、データ領域はクラスタ単位で管理され、そのクラスタが属しているファイルとクラスタとのつながりの順序、あるいは未使用クラスタであるといった情報はFAT領域302によって管理されている。したがって、あるファイルに属するクラスタの集合とクラスタとのつながりの順序はFAT領域302によって論理的に連続してつながっていることになり、データ領域304においては、必ずしも一つのファイルのデータが物理的に連続してい

10

20

30

40

50

るとは限らない。また、1つのFAT領域の大きさを示すFATセクタ数は、ファイルの処理の対象のディスクに最初にアクセスがあった時点で、ブートセクタ301からその値を求め、すでにそれがRAM202上に記憶されているものとする。

【0023】

図7は、本実施形態におけるFATの更新処理手順の一例を示すフローチャートである。

まず、ステップS701において、FAT処理部105の制御によりバッファ制御部106は、これから更新するFAT項目の存在するセクタの番号を、図6のセクタ番号602に記憶する。なお、ディスク108上に図3に示すようなFAT1、FAT2の両方に存在するが、ここではFAT1の範囲にある該当のセクタ番号を記憶する。

10

【0024】

次に、ステップS702において、バッファ制御部106は、ステップS701で記憶したセクタ番号のデータを図6のデータ603に書き込む。そして、ステップS703において、バッファ制御部106は、ステップS702で書き込まれたバッファ上で、指示通りに該当するFAT項目を変更する。そして、ステップS704において、バッファ制御部106は、変更を行ったことを示すため、状態ビット601のdirtyビットを立てて処理を終了する。

【0025】

一方、バッファの同期処理はユーザプログラム101から任意のタイミングで指示されることにより適宜行われる。図8は、バッファデータの同期処理の手順の一例を示すフローチャートである。なお、図8に示す各処理は、バッファ制御部106の制御により行われる。

20

まず、ステップS801において、状態ビット601のdirtyビットを参照して、そのバッファがディスク108への書き込みを必要とするセクタのものであるか否かを判断する。ディスク108への書き込みが必要ない場合は、ステップS805に進む。

【0026】

一方、ステップS801の判断の結果、ディスク108への書き込みを必要とするセクタのものである場合は、ステップS802に進む。そして、そのバッファのセクタ番号602を参照して、しかるべきFAT1領域のセクタヘデータ603を書き込むようにディスクドライバ部107へと指示を出す。次に、ステップS803において、あらかじめ求められたFATサイズnから、FAT2の該当するセクタ番号を算出する。具体的には、セクタ番号602にFATサイズnを足したものになる。

30

【0027】

次に、ステップS804において、ステップS802で書き込むよう指示したバッファのデータ603を、ステップS803で求めた対応するFAT2領域のセクタへと書き込むようにディスクドライバ部107へ指示する。そして、ステップS805において、全てのバッファに対し処理が終了したか否かを判断し、終了していない場合はステップS801に戻り、終了した場合はそのまま処理を終了する。以上のように、ステップS801～ステップS804までの処理は、全てのバッファに対して繰り返し行われる。このように、一切FAT2領域の内容をバッファに読み出さず、ファイルへのデータの追加書き込み、削除等のファイル操作に伴って変更されたFATデータを、FAT1及びFAT2の両方へ同じ内容のデータを書き込むことができる。

40

【0028】

なお、本実施形態では、図6に示す構成でRAM202に確保されたバッファはFATデータ専用とした。ところが、このバッファはディレクトリエントリデータやファイルデータ用のバッファと共通のバッファとして用いてもよい。その場合、状態ビット601にFATビットを設け、そのバッファがディレクトリエントリデータやファイルデータとは区別されることを示すこととする。バッファの同期操作の際に、状態ビット601を参照してFATビットが立っている（例えば零以外）場合にのみ、同じバッファのデータ603をFAT2の対応するセクタに書き込むことにより、同様の効果が得られる。

50

【 0 0 2 9 】

(その他の実施形態)

また、本発明は、以下の処理を実行することによっても実現される。即ち、上述した実施形態の機能を実現するソフトウェア（プログラム）を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ（またはCPUやMPU等）がプログラムを読み出して実行する処理である。

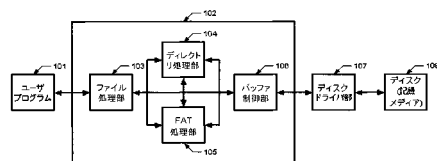
【符号の説明】

【 0 0 3 0 】

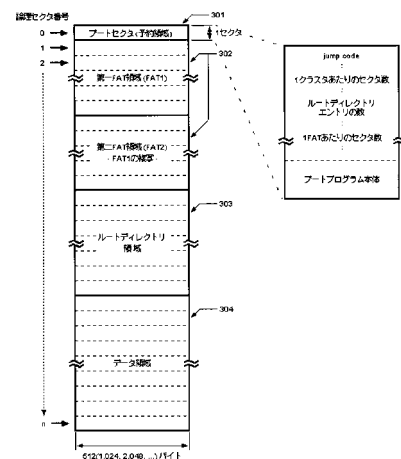
105 FAT処理部、106 バッファ制御部、107 ディスクドライバ部、108 ディスク

10

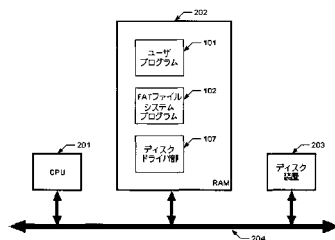
【 図 1 】



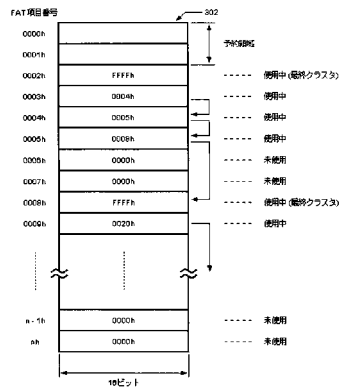
【 図 3 】



【 図 2 】



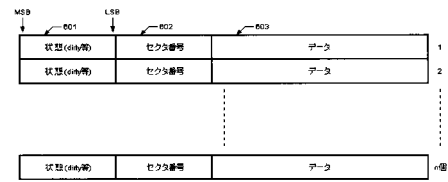
【図 4】



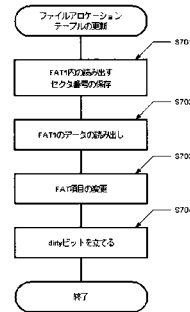
【図 5】



【図 6】



【図 7】



【図 8】

