

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4667014号
(P4667014)

(45) 発行日 平成23年4月6日(2011.4.6)

(24) 登録日 平成23年1月21日(2011.1.21)

(51) Int. Cl.		F 1			
G 0 6 F	12/14	(2006.01)	G 0 6 F	12/14	5 1 0 D
G 0 6 F	21/02	(2006.01)	G 0 6 F	12/14	5 1 0 C
G 0 6 F	21/24	(2006.01)	G 0 6 F	12/14	5 1 0 G

請求項の数 8 (全 26 頁)

(21) 出願番号	特願2004-334805 (P2004-334805)	(73) 特許権者	000001007 キヤノン株式会社 東京都大田区下丸子3丁目30番2号
(22) 出願日	平成16年11月18日(2004.11.18)	(74) 代理人	100126240 弁理士 阿部 琢磨
(65) 公開番号	特開2006-146505 (P2006-146505A)	(74) 代理人	100124442 弁理士 黒岩 創吾
(43) 公開日	平成18年6月8日(2006.6.8)	(72) 発明者	大島 浩義 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
審査請求日	平成19年11月16日(2007.11.16)	(72) 発明者	飯島 克己 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

最終頁に続く

(54) 【発明の名称】 記憶管理装置及びその制御方法

(57) 【特許請求の範囲】

【請求項 1】

記憶媒体のデータ領域におけるデータ単位であるクラスタを用いて、前記記憶媒体のデータ記憶を管理する記憶管理装置であって、

データを、複数のセクタからなり、かつ前記クラスタのサイズとは異なる所定サイズのブロック単位でライトプロテクトするライトプロテクト手段と、

前記ライトプロテクト手段によりライトプロテクトされたブロックへのデータ書き込みを禁止する書き込み禁止手段と、

前記クラスタのサイズと前記クラスタの開始位置を操作することにより、前記クラスタの境界と少なくとも一つ以上の連続した前記ブロックの境界とを一致させるブロック境界一致手段と、を具備することを特徴とする記憶管理装置。

10

【請求項 2】

前記ライトプロテクト手段によりライトプロテクトされたブロックのライトプロテクトを解除するライトプロテクト解除手段をさらに具備することを特徴とする請求項 1 に記載の記憶管理装置。

【請求項 3】

データが格納される少なくとも一つ以上の前記ブロックの全てに対し、データ書き込み後に前記ライトプロテクト手段によりライトプロテクトをかけるライトプロテクト書き込み手段をさらに具備することを特徴とする請求項 1 または 2 に記載の記憶管理装置。

【請求項 4】

20

書き込みデータを連続領域に格納する連続領域格納手段と、

前記ライトプロテクト書き込み手段により、データが格納される少なくとも一つ以上の前記ブロックの全てに対し、前記連続領域格納手段によるデータ書き込み後にライトプロテクトをかける連続領域ライトプロテクト書き込み手段と、をさらに具備することを特徴とする請求項 3 に記載の記憶管理装置。

【請求項 5】

前記ブロック境界一致手段により、前記クラスタの境界と少なくとも一つ以上の前記ブロックの境界が一致している場合において、前記クラスタに一意のアドレスを割り当てるクラスタアドレス割り当て手段と、前記クラスタアドレス割り当て手段により割り当てられたクラスタアドレスにおいて、所定アドレスを持つクラスタを構成する前記ブロック全てに、前記ライトプロテクト手段によりライトプロテクトをかけるクラスタライトプロテクト手段と、をさらに具備することを特徴とする請求項 1 に記載の記憶管理装置。

10

【請求項 6】

データが格納される少なくとも一つ以上のクラスタに対し、データ書き込み後に、前記クラスタライトプロテクト手段によりライトプロテクトをかけるクラスタライトプロテクト書き込み手段をさらに具備することを特徴とする請求項 5 に記載の記憶管理装置。

【請求項 7】

書き込みデータを連続する前記クラスタに格納する連続クラスタ格納手段と、前記クラスタライトプロテクト書き込み手段により、データが格納される少なくとも一つ以上の前記クラスタの全てに対し、前記連続クラスタ格納手段によるデータ書き込み後にライトプロテクトをかける連続クラスタライトプロテクト書き込み手段と、を具備することを特徴とする請求項 5 に記載の記憶管理装置。

20

【請求項 8】

データ記憶領域におけるデータ単位であるクラスタを用いてデータ記憶を管理する記憶管理装置の制御方法であって、

データを、複数のセクタからなり、かつ前記クラスタのサイズとは異なる所定サイズのブロック単位でライトプロテクトするライトプロテクト工程と、

前記ライトプロテクト工程においてライトプロテクトされたブロックへのデータ書き込みを禁止する書き込み禁止工程と、

前記クラスタのサイズと前記クラスタの開始位置を操作することにより、前記クラスタの境界と少なくとも一つ以上の連続した前記ブロックの境界とを一致させるブロック境界一致工程と、を具備することを特徴とする記憶管理装置の制御方法。

30

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、記憶媒体のデータ記憶を管理する記憶管理装置及びその制御方法に関するものである。

【背景技術】

【0002】

近年、固定ディスクや着脱可能記憶媒体、例えば S D (S e c u r e D i g i t a l) メモリカード等が、パーソナルコンピュータ、P D A、携帯電話、デジタルスチルカメラ等で、ファイルを格納するためのストレージデバイスとして用いられている。また、光磁気ディスク等も広く利用されている。

40

【0003】

これら記憶装置は、一度きりの書き込みのみ許すもの、例えば一度データを焼き付けた C D - R 等を除いて、記憶データを物理的に保護することは困難であり、記憶データの保護は大部分がユーザの注意に頼っていた。それは、リード/ライト可能な状態にある記憶装置内データを、特定のソフトウェア(例えば、ファイルシステム等)により管理し、プロテクトしていても、他のソフトウェアからは書換えや削除が可能だからである。特に、リード/ライト可能な着脱可能記憶媒体は、装着した機器によりデータ管理ソフトウェア

50

が決定されるため、ソフトウェア依存のプロテクト方法では、データ保護の実現が難しい。さらに、いずれの記憶装置においても、記憶装置のフォーマットを行うと、データは全て削除されてしまう。

【0004】

しかし近年、データ保護機能を具備する記憶装置が登場してきた。例えばSDメモリーカード等は、記憶媒体のデータ領域全体にライトプロテクトをかける全体ライトプロテクト機能を具備している。これは、記憶媒体の側面のスイッチにより、ライトプロテクトON及びOFFを切り替えることにより、記憶媒体内データ全てをライトプロテクトすることができるという機能である。また、SDメモリーカードは、記憶媒体のデータ領域一部分にライトプロテクトをかける部分ライトプロテクト機能を具備している。これは、記憶媒体内データ領域を、例えば2KBといったWPブロックと呼ばれる単位で分割し、該WPブロック単位でライトプロテクトすることができるという機能である。また、クラスタ単位での記憶データの消去を管理する技術は、特許文献1に開示されている。

10

【特許文献1】特開2001-188701号公報

【発明の開示】

【発明が解決しようとする課題】

【0005】

しかしながら、前記全体ライトプロテクト機能では、一つのファイルを保護するためにも、記憶媒体全体がライトプロテクトしなくてはならないため、非効率かつ使い勝手が悪いという問題があった。

20

【0006】

また、前記部分ライトプロテクト機能では、ファイルの先頭がWPブロックの先頭と一致していなければ、無駄にデータ領域を消費してしまうという問題があった。さらに、例えばFATファイルシステムによりデータを格納する場合、ファイルはクラスタ単位で区切られて、不連続な領域に格納されることがあるため、ファイルの位置を特定するFAT (File Allocation Table) やディレクトリエントリがフォーマットやFAT更新等により変更されてしまうと、ファイルを復旧することが不可能になってしまうという問題もあった。

【0007】

本発明は、記憶媒体に対する無駄のないブロック単位でのライトプロテクト及びその復旧が可能となる記憶管理装置及びその制御方法を提供することを目的とする。

30

【課題を解決するための手段】

【0008】

上記目的を達成するために、本発明の記憶管理装置は、記憶媒体のデータ領域におけるデータ単位であるクラスタを用いて、前記記憶媒体のデータ記憶を管理する記憶管理装置であって、データを、複数のセクタからなり、かつ前記クラスタのサイズとは異なる所定サイズのブロック単位でライトプロテクトするライトプロテクト手段と、前記ライトプロテクト手段によりライトプロテクトされたブロックへのデータ書き込みを禁止する書き込み禁止手段と、前記クラスタのサイズと前記クラスタの開始位置を操作することにより、前記クラスタの境界と少なくとも一つ以上の連続した前記ブロックの境界とを一致させるブロック境界一致手段と、を具備することを特徴とする。

40

【発明の効果】

【0010】

本発明によれば、記憶媒体に対する無駄のないブロック単位でのライトプロテクト及びその復旧が可能となる。

【発明を実施するための最良の形態】

【0011】

以下、図面を参照して本発明の実施の形態を説明する。

【0012】

(第1の実施形態)

50

図1は第1の実施形態におけるカメラシステムの構成を表すブロック図である。本カメラシステムは、電子カメラと、これに着脱可能な外部記憶媒体として、SDメモリカード17、PC通信インターフェース19、及びPC通信インターフェース19を介して電子カメラと通信可能に接続されたパーソナルコンピュータ22から構成される。

【0013】

1はレンズであり、2はレンズ1を通った光を電気信号として出力するCCDユニットである。3はA/Dコンバータであり、CCDユニット2からのアナログ信号をデジタル信号へ変換する。4はSSGユニットであり、CCDユニット2とA/Dコンバータ3に同期信号を供給する。5はCPUであり、本カメラシステムにおける各種の制御を実現する。

10

【0014】

6は信号処理アクセラレータであり、信号処理を高速に実現する。7は電池であり、8は電池7よりの電力を電子カメラ全体へ供給するためのDC/DCコンバータである。9は電源コントローラユニットであり、DC/DCコンバータ8をコントロールする。10はパネル操作・教示装置・電源のコントロールを行うサブCPUである。11はユーザへ各種の情報を表示する装置であり、液晶パネル等が用いられる。12はコントロールパネルであり、ユーザが直接操作するリリーススイッチ、及びメニューボタンを含む。

【0015】

13はROMであり、OS等のシステムプログラムを格納する。14はDRAMであり、本電子カメラの主記憶であり。15はフラッシュROMであり、内蔵記憶媒体として使用する。16はSDカード等のカーとインターフェース部、17はSDメモリカードの外部記憶媒体である。また、18はDMAコントローラ、19はストロボである。

20

【0016】

この電子カメラの撮影時の動作について説明する。コントロールパネル12のリリーススイッチをユーザが押すと、CPU5がそのことを検出して撮影シーケンスを開始する。以下の動作は全てCPU5によるコントロールで行われることを前提とする。

【0017】

さて、リリーススイッチの押下により、SSG4がCCD2を駆動する。CCD2から出力されるアナログ信号は、A/Dコンバータ3でデジタル信号へ変換される。A/Dコンバータ3の出力は、DMAコントローラ18によってDRAM14へDMA転送される。1フレーム分のDMA転送が終了した時点でCPU5は、信号シーケンスを開始する。

30

【0018】

信号処理シーケンスでは、フラッシュROM15から信号処理プログラムを主記憶(DRAM14)上に読み出し、主記憶上のデータを信号処理アクセラレータ6へ転送し信号処理を行う。ただし、信号処理アクセラレータ6は信号処理の全てを行うわけではなく、CPU5で行う処理の特に時間のかかる処理などを助ける演算回路であり、CPU5の処理ソフトウェアと連携して動作する。信号処理の一部または全部が終了すると画像ファイルとして、カードインターフェース16を介して、SDメモリカード17へ記録する。このとき、CPU5もしくは信号処理アクセラレータ6は、記録するファイルフォーマットが圧縮処理を必要とするのであれば画像ファイルの圧縮も行う。

40

【0019】

図2は、図1に示したSDメモリカード17に記録される画像ファイルシステムの階層構造を示す概念図であり、例えば階層L1~L4で構成される。

【0020】

図2において、最上位の階層L1はユーザアプリケーションで、電子カメラ内部で起動するソフトウェアであり、ファイルをファイル名でオープンして読み書きした後クローズする。階層L2は、ファイルシステムAPI層で、ユーザアプリケーションからファンクションコールによって呼び出される。階層L2がドライブ名とファイルシステムを関連付けて管理している。各ドライブ毎にファイルシステムアーキテクチャ層(階層L3)をマウントするように構成しているため、複数のファイルシステムアーキテクチャ(例えばF

50

A T、H P - F S、N F S、N T F S等)を混在されることが可能となっている。なお、ファイルシステムアーキテクチャ層が実際のファイル管理を行う部分である。最下位の階層L 4がブロックデバイス層である。

【0021】

ファイルシステムアーキテクチャ層がブロックデバイス層の提供するサービスを利用してファイル入出力を実現している。なお、ブロックデバイス層では、データを例えば512バイト等のセクタという単位で管理している。該ブロックデバイス層でデバイス毎の入出力制御の違いと、ヘッドやシリンダ等のパラメータの違いを吸収している。これにより、同時に複数の種類のデバイスを混在させたカメラシステムを構築することが出来る。

【0022】

第1の実施形態におけるカメラシステムでは、SDメモ리카ード17のブロックデバイスがファイルシステムへ提供するサービスは、(A.1)論理的なセクタ番号で指定したセクタからのデータの読み出し、ならびに(A.2)論理的なセクタ番号で指定したセクタへのデータの書き込みである。

【0023】

ここで、記憶媒体のデータ構造及びFATファイルシステムについて説明する。図5は、記憶媒体のデータ構造を示している。図5において、501はMBR(Master Boot Record)であり、記憶媒体の先頭から512バイトのデータサイズを占め、図6に示す構造を持っている。該記憶媒体が起動可能な記憶媒体である場合は、601のブートストラップローダ、該記憶媒体の起動プログラムを所持している。起動可能ではない場合は、ブートストラップローダを持っていない。ブートストラップローダ601は、MBR501の先頭から446バイトまでを占める。602はパーティションテーブルであり、該記憶媒体のパーティション502~50Nの情報が収められ、MBR501の447バイト目から64バイト、510バイト目までに格納されている。

【0024】

パーティションとは、該記憶媒体の区分けを意味し、一つのパーティションは、一つのファイルシステムにより管理される。一つの記憶媒体は、少なくとも一つ以上のパーティションによって構成される。パーティションテーブル602には、各パーティションの物理的な開始位置、終了位置、論理的な開始位置、総セクタ数が収められている。例外的にパーティションが一つの場合、MBRを持たないことがある。その場合は、記憶媒体全体を一つのパーティションとみなす。

【0025】

続いて、FATファイルシステムで論理フォーマットされたパーティションのデータ構造を説明する。図7は、FAT16ファイルシステムによりフォーマットした場合のデータ構造を示している。701はPBR(Partition Boot Record)で、図8に示す構造を持っている。図8の801はジャンプ命令で、IPL(Initial Program Loader)803へのジャンプ命令が格納されている。802はBPB(BIOS Parameter Block)でファイルシステムが扱う該パーティションに特化した情報が記録されている。図9は該BPBに含まれるパラメータを示している。803はIPLでこのパーティションの起動プログラムが収められている。

【0026】

図7の702、及び703はFAT(File Allocation Table)で、ユーザデータ領域705に格納されているファイルの位置情報が収められている。FATファイルシステムは、FAT702及び703により、パーティションに格納されているファイルの位置情報を取得する。そのため、ユーザデータ領域にファイルが格納されていても、FAT702及び703に該ファイルの位置情報が格納されていなければ、ファイルとして認識できない。FAT702及び703には、全く同じデータが収められており、片方はバックアップとして利用する。704はルートディレクトリエントリで、ルートディレクトリに格納されたファイルの管理情報が収められている。705はユーザデ

10

20

30

40

50

ータ領域でファイルの本体が格納されている。

【 0 0 2 7 】

第 1 の実施形態における S D メモリカード 1 7 は、前記 M B R を持たず、データ領域全体が一つのパーティションを構成し、該パーティションは、F A T 1 6 ファイルシステムによりフォーマットされ、管理されている。ここで、S D メモリカード 1 7 に対するファイルの書き込み、読み出し動作について説明する。

【 0 0 2 8 】

S D メモリカード 1 7 内のデータが、図 1 0 に示す状態になっているものとする。図 1 0 において、1 0 0 1 はルートディレクトリであり、1 0 0 2 はディレクトリ、1 0 0 3、1 0 0 4 及び 1 0 0 5 はファイルである。ディレクトリ 1 0 0 2、ファイル 1 0 0 4 及び 1 0 0 5 は、ルートディレクトリ 1 0 0 1 に格納されている。また、ファイル 1 0 0 3 は、ディレクトリ 1 0 0 2 に格納されている。

10

【 0 0 2 9 】

このときの S D メモリカード 1 7 内データ構造を、図 1 1 に示す。図 1 1 において、図 1 0 と同一のものは、同じ番号を振っている。ただし、ディレクトリはデータ構造においてはディレクトリエントリと呼ぶ。1 1 0 1、1 1 0 2、及び 1 1 0 3 は、エンTRIES アイテムであり、ルートディレクトリ 1 0 0 1 内に格納されているディレクトリ 1 0 0 2 及びファイル 1 0 0 4、1 0 0 5 の格納場所を示している。同じくエンTRIES アイテム 1 1 0 4 は、ディレクトリ 1 0 0 2 に格納されているファイル 1 0 0 3 の格納場所を示している。

【 0 0 3 0 】

ディレクトリエントリ及びファイルは、前記 B P B 内パラメータ、クラスタサイズ (図 9 の 9 0 3) が示すサイズ、例えば 8 K B 等の単位で格納されている。ディレクトリエントリ、及びファイルは、1 クラスタ以上のデータ量を持つ場合は、複数のクラスタにまたがって格納される。クラスタの使用状況、及びクラスタ同士のリンク情報は、前記 F A T に格納されている。ファイル 1 0 0 3 は、クラスタ 1 1 1 1、及び 1 1 1 2 にまたがって格納されている。連続したクラスタに格納されるか、不連続のクラスタに格納されるかは、S D メモリカード 1 7 内のクラスタ使用状況によって決定される。

20

【 0 0 3 1 】

ファイルデータを読み出すファイル読み出し手段を、ディレクトリ 1 0 0 2 に格納されている、ファイル 1 0 0 3 を読み出す場合の動作を例にとり説明する。データの読み出し、及び書き込み命令は、全てファイルシステムソフトウェアが、前記サービス (A . 1) 及び (A . 2) を利用して行う。ただし、ディレクトリ 1 0 0 2 のディレクトリエントリが格納されているクラスタ番号はわかっているものとする。

30

【 0 0 3 2 】

図 1 2 に前記ファイル読み出し手段をフローチャートで示している。まず、ファイルシステムソフトウェアは、ステップ S 1 2 0 1 でディレクトリ 1 0 0 2 のクラスタをカレントクラスタとし、ステップ S 1 2 0 2 でカレントクラスタの先頭から、ファイル 1 0 0 3 のエンTRIES アイテムを検索する。続いて、ステップ S 1 2 0 3 でエンTRIES アイテムを発見したかを判断し、発見できなかった場合は、ステップ S 1 2 0 4 でカレントクラスタのリンク情報を前記 F A T から取得する。該リンク情報を基に、ステップ S 1 2 0 5 でカレントクラスタがリンクされているかを判断する。リンクされていた場合、ステップ S 1 2 0 6 で、リンク先クラスタを新たなカレントクラスタとし、ステップ S 1 2 0 2 へ戻り、繰り返す。

40

【 0 0 3 3 】

ステップ S 1 2 0 3 で、ファイル 1 0 0 3 のエンTRIES アイテムを発見した場合は、ステップ S 1 2 0 7 で該エンTRIES アイテムのリンク先クラスタの情報を前記 F A T から読み出す。該情報は、該クラスタの使用状況及び、リンク状況である。該情報を基に、ステップ S 1 2 0 8 で該クラスタの使用状況を確認する。該クラスタが使用されている場合は、ステップ S 1 2 0 9 で該クラスタのデータを読み出し、ステップ S 1 2 1 0 で該クラスタがリンクされているかを判断する。リンクされていると判断した場合は、ステップ S 1 2 0

50

7に戻り、読み出し動作を繰り返す。リンクされていないと判断した場合は、正常終了する。ステップS1205でリンクされていなかった場合、及びステップS1208で未使用と判断した場合は、ファイルが存在しない、もしくはファイルが壊れていると判断し、エラー終了する。

【0034】

ファイルデータを書き込むファイル書き込み手段を、ディレクトリ1002に対し、ファイルを作成し、書き込みを行う場合の動作を例にとり説明する。データの読み出し、及び書き込み命令は、全てファイルシステムソフトウェアが、前記サービス(A.1)及び(A.2)を利用して行う。ただし、ディレクトリ1002のディレクトリエントリが格納されているクラスタ番号はわかっているものとする。

10

【0035】

図13に前記ファイル書き込み手段をフローチャートで示している。ステップS1301でディレクトリ1002のディレクトリエントリが格納されたクラスタをカレントクラスタとし、ステップS1302でカレントクラスタのディレクトリエントリアイテムを先頭から順に読み出し、エントリアイテムの空きを検索する。ステップS1303で発見したかを判断し、発見できなかった場合は、ステップS1304でカレントクラスタのリンク情報を取得する。該リンク情報を基に、ステップS1305でカレントクラスタがリンクされているかを判断する。リンクされていた場合、ステップS1306で、リンク先クラスタを新たなカレントクラスタとし、ステップS1302へ戻る。

【0036】

20

ステップS1305でリンクされていなかった場合は、新たに空きクラスタを、ディレクトリ1002のディレクトリエントリ領域としてリンクさせ拡張する。先ず、ステップS1307で、前記FATを先頭から順に読み出し、空きクラスタを検索する。ステップS1308で空きクラスタを発見したかを判断し、発見した場合は、ステップS1309でカレントクラスタを発見した空きクラスタへリンクし、空きクラスタはリンクしないようにFATを修正する。さらに、カレントクラスタを発見した空きクラスタに変更する。

【0037】

ステップS1303でエントリアイテムの空きを発見した場合、及びステップS1309でFATを修正した場合は、ステップS1310でファイルデータ書き込み領域の有無を調べるため、前記FATを先頭から順に読み出し、空きクラスタを検索する。さらに、ステップS1311で、空きクラスタを発見したかを判断し、発見した場合は、ステップS1312で該クラスタへのリンク情報を持ったエントリアイテムを、カレントクラスタのディレクトリエントリに対し追加書き込みを行う。ステップS1309で新たなクラスタをリンクさせた場合は、カレントクラスタの先頭にエントリアイテムを書き込む。続いて、ステップS1313で該クラスタにファイルデータを1クラスタ分書き込み、ステップS1314でまだ書き込まれていないファイルデータがあるかを判断する。まだファイルデータがあった場合は、ステップS1315で再び前記FATから空きクラスタを検索する。ステップS1316で空きクラスタを発見したかを判断し、発見した場合は、ステップS1313に戻り書き込み動作を繰り返す。ステップS1314で、もう全てのファイルデータの書き込みが完了したと判断した場合、ステップS1317で、ファイルデータを書き込んだクラスタを順にリンクさせ、最後のクラスタはリンクさせないように、前記FATの更新を行い、正常終了する。ステップS1308、ステップS1310、及びステップS1315で空きクラスタを発見できなかった場合は、エラー終了する。

30

40

【0038】

ファイルデータを削除するファイル削除手段を、ディレクトリ1002内のファイル1003を削除する場合の動作を例にとり説明する。データの読み出し、及び書き込み命令は、全てファイルシステムソフトウェアが、前記サービス(A.1)及び(A.2)を利用して行う。ただし、ディレクトリ1002のディレクトリエントリが格納されているクラスタ番号はわかっているものとする。

【0039】

50

図14に前記ファイル削除手段をフローチャートで示している。まず、ファイルシステムソフトウェアは、ステップS1401でディレクトリ1002のクラスタをカレントクラスタとし、ステップS1402でカレントクラスタの先頭から、ファイル1003のエントリアイテムを検索する。続いて、ステップS1403でエントリアイテムを発見したかを判断し、発見できなかった場合は、ステップS1404でカレントクラスタのリンク情報を前記FATから取得する。該リンク情報を基に、ステップS1405でカレントクラスタがリンクされているかを判断する。リンクされていた場合、ステップS1406で、リンク先クラスタを新たなカレントクラスタとし、ステップS1402へ戻り、繰り返す。ステップS1403で、ファイル1003のエントリアイテムを発見した場合は、ステップS1407で該エントリアイテムに無効データを書き込み、ステップS1408でFATを更新し、ファイル1003が使用していたクラスタ全てを未使用領域にし、正常終了する。ステップS1405でリンクされていなかった場合は、ファイルが存在しないと判断し、エラー終了する。

10

【0040】

以下、SDメモリカードのライトプロテクト機能について説明する。SDメモリカードには、WPブロックと呼ばれる単位でデータ領域の一部をライトプロテクトする機能が具備されている。WPブロックのサイズは、1セクタ～2の14乗セクタまでの値がとられ、セクタサイズとともに、カード固有の値となっている。WPブロックは、図19に示すように、データ領域の先頭から重なることなく固定で割り付けられ、各WPブロックには、一意にWPブロックアドレスが割り当てられている。

20

【0041】

WPブロックに対する操作には、(B.1)指定WPブロックにライトプロテクトをかける操作、(B.2)指定WPブロックにかけられているライトプロテクトを解除する操作、ならびに(B.3)指定WPブロックがライトプロテクトされているかの状態を参照する操作がある。各操作は、SDカードドライバがSDメモリカードへコマンドを送信し、該コマンドをSDメモリカード内コントローラが処理することで行われる。各操作について説明する。

【0042】

図15は、前記(B.1)の操作をフローチャートで示した図である。まず、ステップS1501で前記SDカードドライバは、SDメモリカード17へライトプロテクトを要求するコマンドを送信する。該コマンド内には、ライトプロテクトをかけるWPブロックのWPブロックアドレスが含まれている。該コマンドを受信したSDメモリカード内コントローラは、ステップS1502で、指定WPブロックアドレスが有効な範囲内であるかを判定する。有効な範囲内であると判定した場合は、ステップS1503で指定WPブロックアドレスのWPブロックが現在ライトプロテクトされているかを判定する。現在ライトプロテクトされていないと判定した場合は、ステップS1504で指定WPブロックアドレスのWPブロックにライトプロテクトをかけ、正常終了する。ステップS1503で現在ライトプロテクトされていると判定した場合は、その時点で正常終了する。ステップS1502で指定WPブロックアドレスが有効な範囲外であると判定した場合は、エラー終了する。

30

40

【0043】

図16は、前記(B.2)の操作をフローチャートで示した図である。まず、ステップS1601で前記SDカードドライバは、SDメモリカード17へライトプロテクト解除を要求するコマンドを送信する。該コマンド内には、ライトプロテクトを解除するWPブロックのWPブロックアドレスが含まれている。該コマンドを受信したSDカード内コントローラは、ステップS1602で、指定WPブロックアドレスが有効な範囲内であるかを判定する。有効な範囲内であると判定した場合は、ステップS1603で指定WPブロックアドレスのWPブロックが現在ライトプロテクトされているかを判定する。現在ライトプロテクトされていると判定した場合は、ステップS1604で指定WPブロックアドレスのWPブロックのライトプロテクトを解除し、正常終了する。ステップS1603で

50

現在ライトプロテクトされていないと判定した場合は、その時点で正常終了する。ステップS 1 6 0 2で指定WPブロックアドレスが有効な範囲外であると判定した場合は、エラー終了する。

【 0 0 4 4 】

図17は、前記(B.3)の操作をフローチャートで示した図である。まず、ステップS 1 7 0 1で前記SDカードドライバは、SDメモリカード17へライトプロテクト状態参照を要求するコマンドを送信する。該コマンド内には、ライトプロテクト状態を参照するWPブロックのWPブロックアドレスが含まれている。該コマンドを受信したSDカード内コントローラは、ステップS 1 7 0 2で、指定WPブロックアドレスが有効な範囲内であるかを判定する。有効な範囲内であると判定した場合は、ステップS 1 7 0 3で指定WPブロックアドレスのWPブロックが現在ライトプロテクトされているかを判定し、その結果をレスポンスに載せ、返信する。ステップS 1 7 0 3における判定がいずれの場合でも、正常終了する。ステップS 1 7 0 2で指定WPブロックアドレスが有効な範囲外であると判定した場合は、エラー終了する。

10

【 0 0 4 5 】

以下、第1の実施形態におけるSDメモリカード17のデータ構造について説明する。図18にSDメモリカード17の初期状態を示している。図18において、1801はPBR、1802はFAT、1803はルートディレクトリエントリである。また、1804はプロテクトファイル格納用のディレクトリエントリであるプロテクトディレクトリエントリで、ルートディレクトリエントリの直ぐ後に位置するクラスタである。該クラスタは、クラスタ番号、『2』番となる。該プロテクトディレクトリエントリには、ライトプロテクトがかけられている。また、該プロテクトディレクトリエントリはリンクすることはない。そのため、プロテクトディレクトリに格納できるファイル数は制限される。1805は空きクラスタである。該クラスタは、クラスタ番号『3』番であり、前記FATを修正し、使用済みクラスタとなっている。該クラスタは、後述するプロテクトファイル復旧手段において、クラスタのサイズを特定するために利用する。

20

【 0 0 4 6 】

以下、第1の実施形態における、本発明の第1の特徴であるWPブロック境界一致手段について説明する。該WPブロック境界一致手段とは、クラスタ単位でライトプロテクトを実現するため、クラスタサイズを前記WPブロックサイズの整数倍のサイズにし、かつクラスタ境界と前記WPブロック境界を一致させる手段である。前記カメラシステムでSDメモリカード17をフォーマットすると、該WPブロック境界一致手段により、クラスタ境界と前記WPブロック境界が一致した状態でフォーマットされる。SDメモリカード17は、前記カメラシステムでフォーマットされているものとする。

30

【 0 0 4 7 】

クラスタ境界は、クラスタ開始位置とクラスタサイズによって決定する。一方WPブロックは、SDメモリカード17固有のサイズであり、可変ではない。そのため、クラスタの開始位置とクラスタサイズをWPブロックに合わせることで境界を一致させる。クラスタ開始位置は、ルートディレクトリエントリの直後であり、図23に示す式で決定する。図23において、2301はパーティションの開始アドレスであり、本実施形態においては先頭アドレス『0』である。2302はリザーブセクタ数、2303はセクタサイズ、2304はFATサイズ、2305はFATの数、ならびに2306はルートディレクトリエントリ数であり、前記BPBのパラメータに含まれている。このとき、図23の式で導かれる値、クラスタ開始位置アドレスが前記WPブロックサイズの整数倍になるようにパラメータを当てはめる。さらに、クラスタサイズは、前記BPBのパラメータであるため、クラスタサイズが前記WPブロックの整数倍になるように、パラメータを当てはめる。前記カメラシステムは、このようにして作成したBPBを基にフォーマットを行うことで、クラスタ境界と前記WPブロックの境界を一致させた状態でのフォーマットを実現している。

40

【 0 0 4 8 】

50

具体的な数値を挙げて説明する。SDメモ리카ード17の固有パラメータが、図20に示す値であるとき、クラスタサイズをWPブロックサイズの4倍の8Kバイトとし、クラスタ境界とWPブロック境界を一致させた上で、図18に示す初期状態を生成する。

【0049】

まず、データ領域を無駄にしないため、できる限り多くのクラスタを生成することを目指す。データ領域全体をクラスタとして割り当てると考えると、10M ÷ 8Kから1280個のクラスタが割り当てられることがわかる。しかし、先頭にリザーブセクタ数(PBRを含む)、FATサイズ(×2)、及びルートディレクトリエントリサイズが必要であるため、この分を差し引いた上での最大数を求める。

【0050】

まず、リザーブセクタ数は、最低でPBR領域として1セクタ必要である。該リザーブセクタ数をYとする。FATサイズは、クラスタ数分のFATエントリを格納できるだけの領域を2個分(FATの数)持つ必要がある。一つのFATエントリは、FAT16においては、16ビットであるため、1セクタで256個のFATエントリを持つことができることがわかる。FATは、少なくともクラスタ数以上のFATエントリを持つ必要があるが、その条件さえ満たしていれば、できるだけ小さい方が良い。一つのFATに必要なセクタ数をXとする。ルートディレクトリエントリサイズは、ルートディレクトリエントリ数に依存する。ルートディレクトリエントリ数は通常512であり、1ディレクトリエントリが32バイトであるため、ルートディレクトリエントリサイズは16Kバイト(32セクタ)である。また、クラスタ境界とWPブロック境界を一致させるため、リザーブセクタ数、FAT、及びルートディレクトリエントリを足し合わせた値が、WPブロックの整数倍になるようにしなければならない。

【0051】

以上の条件を基に式を立てる。

(式1) : クラスタ最大数 - (リザーブセクタ数 + FATセクタ数 × 2 + ルートディレクトリエントリサイズ) / (クラスタサイズ / セクタサイズ) < 1セクタのFATエントリ数 × FATセクタ数

(式2) : (リザーブセクタ数 + FATセクタ数 × 2 + ルートディレクトリエントリサイズ) = (WPブロックサイズ / セクタサイズ)

(式1)、(式2)に値を代入すると以下ようになる。

(式3) : 1280 - (Y + 2X + 32) / 16 < 256X (X、Yは整数)

(式4) : Y + 2X + 32 = 4 (X、Y、 は任意の整数)

さらに、(式3)を変換すると、以下ようになる。

(式5) : 4098X > 20480 - Y

(式5)において、XはFATのセクタ数であるため、最小値を取る必要がある。Xは整数であるため、Xは5を取ることがわかる。

【0052】

これをもとに、(式4)を変換すると、以下ようになる。

(式6) : Y + 42 = 4

(式6)において、Yはリザーブセクタ数であり、1以上の整数であれば良い。無駄を省くため最小値を取ると、Yは2であることがわかる。

【0053】

以上の計算により、BPBパラメータを図21に示すように設定した上で、フォーマットを行うと、図22に示すようなデータ構造となり、クラスタ境界とWPブロック境界を一致させることができる。このWPブロック境界一致手段により、クラスタ単位のライトプロテクトが可能となり、ひいてはディレクトリエントリ及びファイル単位のライトプロテクトが可能となる。

【0054】

以下、先に説明したSDメモ리카ードのライトプロテクト機能を用い、第1の実施形態における、本発明の第2の特徴であるライトプロテクトアクセス手段について説明する。

10

20

30

40

50

該ライトプロテクトアクセス手段とは、SDメモリカード17に対する、ライトプロテクトされたファイルの書き込みを行うライトプロテクト書き込み手段、ライトプロテクトされたファイルの削除を行うライトプロテクト削除手段、並びにライトプロテクトされたファイルの書き換えを行うライトプロテクト書き換え手段によって構成される。

【0055】

先ず、前記ライトプロテクト書き込み手段について説明する。該ライトプロテクト書き込み手段とは、書き込み対象のファイルが新規書き込みファイルである場合に、該ファイルにライトプロテクトをかけ書き込みを行う手段である。図3は、SDメモリカード17に対し、該ライトプロテクト書き込み手段によりファイルを書き込むときの動作をフローチャートで示している。読み出し、及び書き込み命令は、全てファイルシステムソフトウェアが、前記サービス(A.1)及び(A.2)を利用して行う。また、プロテクト実行/解除、状態参照命令は、前記サービス(B.1)~(B.3)を利用して行う。ただし、クラスタ(ディレクトリエントリを含む)に対するライトプロテクト実行/解除、状態参照は、該クラスタを構成する全ての前記WPブロックに対して、該サービス(B.1)~(B.3)を実行しているものとする。

10

【0056】

図3のフローチャートを説明する。先ず、ステップS301で前記プロテクトディレクトリエントリを先頭から順に読み出し、エントリアイテムの空きを検索する。ステップS302で発見したかを判断し、発見した場合は、続いてステップS303でファイルデータを格納するのに必要なクラスタ数を計算する。さらに、ステップS304でFATを読み出し、必要なクラスタ数分の連続した空きクラスタを検索し、ステップS305で発見したかを判断する。発見した場合は、ステップS306で前記プロテクトディレクトリエントリのライトプロテクトを解除し、ステップS307で該連続空きクラスタの先頭クラスタへのリンク情報をもったエントリアイテムをステップS2501で検索した空きエントリアイテム位置に格納し、ステップS308で再度該プロテクトディレクトリエントリにライトプロテクトをかける。さらに、ステップS309でファイルデータを該連続空きクラスタに書き込み、ステップS310で該連続空きクラスタにライトプロテクトをかけ、ステップS311でFATを更新し、正常終了する。ステップS302でエントリアイテムの空きを発見できなかった場合と、ステップS305で連続空きクラスタを発見できなかった場合は、エラー終了する。

20

30

【0057】

続いて、前記ライトプロテクト削除手段、及び前記ライトプロテクト書き換え手段について説明する。該ライトプロテクト削除手段とは、前記ライトプロテクト書き込み手段により書き込まれたファイルデータを削除する手段であり、該ライトプロテクト書き換え手段とは、前記ライトプロテクト書き込み手段により書き込まれたファイルデータを書き換え、書き込んだファイルに再度ライトプロテクトをかける手段である。該ライトプロテクト削除手段、及び該ライトプロテクト書き換え手段はともに、最初に現在書き込まれているファイルデータのライトプロテクトを解除するライトプロテクト解除動作は共通している。該ライトプロテクト解除動作後、前記ライトプロテクト削除手段は前記サービス(A.3)により削除を行い、前記ライトプロテクト書き換え手段は前記ライトプロテクト書き込み手段により書き込みを行い、動作を完了する。

40

【0058】

前記ライトプロテクト解除動作について説明する。図24は、SDメモリカード17内に存在するライトプロテクトされたファイルのライトプロテクトを解除するときの動作をフローチャートで示している。読み出し、及び書き込み命令は、全てファイルシステムソフトウェアが、前記サービス(A.1)及び(A.2)を利用して行う。また、プロテクト実行/解除、状態参照命令は、前記サービス(B.1)~(B.3)を利用して行う。ただし、クラスタ(ディレクトリエントリを含む)に対するライトプロテクト実行/解除、状態参照は、該クラスタを構成する全ての前記WPブロックに対して、該サービス(B.1)~(B.3)を実行しているものとする。

50

【 0 0 5 9 】

図 2 4 のフローチャートを説明する。まず、ステップ S 2 4 0 1 で前記プロテクトディレクトリエントリを先頭から順に読み出し、書き換え及び削除対象ファイルのエントリアイテムを検索する。ステップ S 2 4 0 2 で発見したかを判断し、発見した場合は、続いて、ステップ S 2 4 0 3 で該エントリアイテムから、書き換え及び削除対象ファイルデータが格納されている先頭クラスタを取得し、ステップ S 2 4 0 4 で該クラスタをカレントクラスタとし、ステップ S 2 4 0 5 でカレントクラスタのライトプロテクトを解除する。続いて、ステップ S 2 4 0 6 で、前記 F A T からカレントクラスタのリンク情報を取得する。該リンク情報を基に、ステップ S 2 4 0 7 でカレントクラスタがリンクされているかを判断する。リンクされていた場合、ステップ S 2 4 0 8 で、リンク先クラスタを新たなカレントクラスタとし、ステップ S 2 4 0 5 へ戻る。ステップ S 2 4 0 7 で、リンクされていなかった場合は正常終了する。ステップ S 2 4 0 2 で、エントリアイテムを発見できなかった場合は、エラー終了する。

10

【 0 0 6 0 】

以下、第 1 の実施形態における、本発明の第三の特徴であるライトプロテクトファイル復旧手段について説明する。該ライトプロテクトファイル復旧手段とは、フォーマット等により、S D メモリカード 1 7 のファイルフォーマットが破壊されてしまった場合に、ユーザもしくはシステムの指示により、前記ライトプロテクト書き込み手段により書き込んだファイルを復旧する手段である。ここで、ファイルフォーマットが破壊されてしまった場合とは、S D メモリカード 1 7 が、別のホスト（例えば P C 等）でフォーマットされ、B P B パラメータや F A T が書き換えられ、フォーマットされる前のファイル格納位置を特定することができなくなってしまった場合を指す。

20

【 0 0 6 1 】

該ライトプロテクトファイル復旧手段は、情報取得ステップとファイルデータ取得ステップにより実現する。該情報取得ステップでは、前記ライトプロテクトファイル書き込み手段により書き込んだファイルのデータが格納されている位置を特定するための情報を取得するステップであり、該ファイルデータ取得ステップは、情報取得ステップで取得したファイル位置情報を基に、ファイルデータを取得するステップである。

【 0 0 6 2 】

まず、本実施形態における前記情報取得ステップについて説明する。本実施形態においては、前記ライトプロテクト書き込み手段により書き込まれたファイルは、連続したクラスタに書き込まれている。そのため、該ファイルの位置を特定するためには、クラスタの開始位置とサイズ、及び該ファイルの開始クラスタとファイルのサイズを特定する必要がある。本実施形態においては、図 1 8 に示すように、前記プロテクトディレクトリエントリを 1 つのクラスタで実現し、該プロテクトディレクトリエントリは、初期状態でクラスタ番号『 2 』番の位置に配置させている。さらに、該プロテクトディレクトリエントリの直後のクラスタは、初期状態で使用済みとし、ライトプロテクトをかけていない。さらに、該プロテクトディレクトリエントリの前には、プロテクトされた領域を作成しないことを前提としていた。そのため、データ領域の先頭からプロテクトされた W P ブロックを検索すれば、最初に見つかった W P ブロックが該プロテクトディレクトリエントリの先頭であることがわかり、さらに、該 W P ブロックから連続でプロテクトされている領域を検索すれば、最後に見つかった W P ブロックが該プロテクトディレクトリエントリの終端であることがわかる。これにより、クラスタサイズと、クラスタ開始位置が特定できる。また、プロテクトディレクトリエントリには、プロテクトされたファイルのエントリアイテムが格納されているため、ファイルデータの開始クラスタとサイズを特定することができる。

30

40

【 0 0 6 3 】

続いて、本実施形態における前記ファイルデータ取得ステップについて説明する。本実施形態においては、前記ライトプロテクト書き込み手段により書き込まれたファイルは、連続したクラスタに書き込まれている。そのため、前記情報取得ステップで取得したクラ

50

スタの開始位置とサイズ、及びファイルの開始クラスタとサイズにより、ファイルデータを取得することができる。

【0064】

図4は、該ライトプロテクトファイル復旧手段の動作をフローチャートで示している。図4を説明する。まず、ステップS401でSDメモリカード17のデータ領域の先頭から順に、ライトプロテクトされているWPブロックを検索し、ステップS402で発見したかを判断する。発見した場合には、ステップS403で連続して何個のWPブロックがライトプロテクトされているかを調べる。これにより、前記プロテクトディレクトリエン트리と、ファイルフォーマットが破壊される前のクラスタサイズが確定する。

【0065】

続いて、ステップS404で該プロテクトディレクトリエントリアイテムを検索する。ステップS405で発見したかを判断し、発見した場合には、ステップS406でエントリアイテムから、該エントリアイテムが示すファイルの開始クラスタ番号と、サイズを取得する。また、前記プロテクトディレクトリエントリのクラスタ(クラスタ番号『2』)を基準にして、ファイルデータ開始アドレスを特定する。次に、ステップS407でファイルデータ開始アドレスから、ファイルサイズ分のデータ領域がライトプロテクトされているかを判断する。該データ領域がライトプロテクトされていた場合、ステップS408でファイルデータを読み出し、他の記憶媒体に保存する。さらに、ステップS409で該データ領域のライトプロテクトを解除し、ステップS404に戻る。ステップS405で、エントリアイテムを発見できなかった場合は、ステップS410でプロテクトディレクトリエントリのライトプロテクトを解除し、正常終了する。ステップS402で、ライトプロテクトされているWPブロックを発見できなかった場合、ステップS407でライトプロテクトされていないと判断した場合は、エラー終了する。

【0066】

(第2の実施形態)

本発明の第2の実施形態について説明する。本実施形態においても、第1の実施形態で説明したカメラシステムに着脱可能なSDメモリカード17を例に説明を行う。SDメモリカード17のブロックデバイスは、前記サービス(A.1)及び(A.2)を提供可能とし、該SDメモリカード17のデータ領域は、前記MBRを持たず、データ領域全体が一つのパーティションを構成し、該パーティションは、FAT16ファイルシステムによりフォーマットされ、管理されているものとする。また、本実施形態における本発明の第1の特徴であるWPブロック境界一致手段は、第1の実施形態におけるWPブロック境界一致手段と同様の手段であるものとする。

【0067】

以下、第2の実施形態におけるSDメモリカード17のデータ構造について説明する。本実施形態におけるSDメモリカード17のデータ構造は、特殊なフォーマットではなく、図7に示すFATファイルシステムによるフォーマットを用いる。よって、第1の実施形態のように、前記プロテクトディレクトリ等といったプロテクトファイルを格納するためのディレクトリは存在せず、いずれのディレクトリであってもプロテクトされたファイルを格納することができる。また、特殊なフォーマットではないため、第1の実施形態のように、初期状態が限定されるわけではないため、既にファイルの書き込みが行われている状態のSDメモリカードであっても、FATファイルシステムによりフォーマットされていれば、本実施形態を適応することができる。

【0068】

以下、第2の実施形態における、本発明の第2の特徴であるライトプロテクトアクセス手段について説明する。該ライトプロテクトアクセス手段は、ライトプロテクトされたファイルの書き込みを行うライトプロテクト書き込み手段、ライトプロテクトされたファイルの削除を行うライトプロテクト削除手段、並びにライトプロテクトされたファイルの書き換えを行うライトプロテクト書き換え手段によって構成される。

【0069】

先ず、前記ライトプロテクト書き込み手段について説明する。該ライトプロテクト書き込み手段とは、書き込み対象のファイルが新規書き込みファイルである場合に、該ファイルにライトプロテクトをかけ書き込みを行う手段である。該ライトプロテクト書き込み手段は、連続したクラスタにファイルデータを書き込み、該クラスタにライトプロテクトをかける。ただし、ファイルデータが格納される連続クラスタ内、最終クラスタに限り、ライトプロテクトはクラスタ全体ではなく、ファイルデータが格納されているWPブロックのみにかける。このとき、ファイルデータが該最終クラスタを構成するWPブロック全体に及んでいる場合は、該最終クラスタのさらに一つ後のクラスタを使用済み領域として確保する。

【0070】

これにより、該ライトプロテクト書き込み手段により書き込まれたファイルは、ファイルデータが格納されているWPブロックの内、最終WPブロックの直後のWPブロックには、ライトプロテクトがかけられていないことが保証できる。これは、後述する第2の実施形態におけるライトプロテクトファイル復旧手段において、ファイルのサイズを特定するためである。該ライトプロテクト書き込み手段により、二つのファイルが書き込まれ、この二つのファイルデータが連続していた場合、ファイルデータが格納されるクラスタ全てをライトプロテクトしてしまうと、図27のようになり、ライトプロテクトファイル1とライトプロテクトファイル2が連続したライトプロテクト領域に格納され、境界を特定することができない。そこで、最終WPブロック直後のWPブロックには、ライトプロテクトをかけないことで、図28のようになり、ライトプロテクトファイル1、及び2のサイズを特定することができる。

【0071】

図25は、SDメモ리카ード17に対し、該ライトプロテクト書き込み手段によりファイルを書き込むときの動作をフローチャートで示している。読み出し、及び書き込み命令は、全てファイルシステムソフトウェアが、前記サービス(A.1)及び(A.2)を利用して行う。また、プロテクト実行/解除、状態参照命令は、前記サービス(B.1)～(B.3)を利用して行う。ただし、クラスタ(ディレクトリエントリを含む)に対するライトプロテクト実行/解除、状態参照は、該クラスタを構成する全ての前記WPブロックに対して、該サービス(B.1)～(B.3)を実行しているものとする。

【0072】

図25のフローチャートを説明する。先ず、ステップS2501で書き込みを行うディレクトリのディレクトリエントリを先頭から順に読み出し、エントリアイテムの空きを探索する。ステップS2502で発見したかを判断し、発見した場合は、続いてステップS2503でファイルデータを格納するのに必要なクラスタ数を計算する。このとき、ファイルデータが最終クラスタを構成するWPブロックの最終クラスタにまで及ぶ場合は、必要なクラスタ数に1加えた値を必要なクラスタ数とする。さらに、ステップS2504でFATを読み出し、ステップS2503で計算した必要なクラスタ数分の連続した空きクラスタを探索し、ステップS2505で発見したかを判断する。発見した場合は、ステップS2506で該連続空きクラスタの先頭クラスタへのリンク情報をもったエントリアイテムをステップS2501で検索した空きエントリアイテム位置に格納する。さらに、ステップS2507でファイルデータを該連続空きクラスタに書き込み、ステップS2508で該連続空きクラスタの内、最終クラスタを除いた全てのクラスタにライトプロテクトをかける。続いて、ステップS2509で、最終クラスタを構成するWPブロックの内、ファイルデータが格納されたWPブロックに対し、ライトプロテクトをかける。このとき、ファイルデータが格納されたWPブロックが存在しなかった場合、何も行わない。最後に、ステップS2510でFATを更新し、正常終了する。ステップS2502でエントリアイテムの空きを発見できなかった場合と、ステップS2505で連続空きクラスタを発見できなかった場合は、エラー終了する。

【0073】

続いて、前記ライトプロテクト削除手段、及び前記ライトプロテクト書き換え手段につ

10

20

30

40

50

いて説明する。該ライトプロテクト削除手段、及び該ライトプロテクト書き換え手段は、第1の実施形態におけるライトプロテクト削除手段、及びライトプロテクト書き換え手段と同様の動作で実現する。ただし、前記ライトプロテクト解除動作を示す図24のステップS2405において、カレントクラスタが連続クラスタの最終クラスタである場合、該カレントクラスタを構成するWPブロックの内、ライトプロテクトしていないWPブロックが存在する。このとき、該WPブロックの内、ライトプロテクトされているWPブロックのみに対してライトプロテクト解除を行うものとする。

【0074】

以下、第2の実施形態における、本発明の第三の特徴であるライトプロテクトファイル復旧手段について説明する。該ライトプロテクトファイル復旧手段とは、フォーマット等により、SDメモ리카ード17のファイルフォーマットが破壊されてしまった場合に、ユーザもしくはシステムの指示により、前記ライトプロテクト書き込み手段により書き込んだファイルを復旧する手段である。ここで、ファイルフォーマットが破壊されてしまった場合とは、SDメモ리카ード17が、別のホスト(例えばPC等)でフォーマットされ、BPPパラメータやFATが書き換えられ、フォーマットされる前のファイル格納位置を特定することができなくなってしまった場合を指す。

10

【0075】

該ライトプロテクトファイル復旧手段は、情報取得ステップとファイルデータ取得ステップにより実現する。該情報取得ステップでは、前記ライトプロテクトファイル書き込み手段により書き込んだファイルのデータが格納されている位置を特定するための情報を取得するステップであり、該ファイルデータ取得ステップは、情報取得ステップで取得したファイル位置情報を基に、ファイルデータを取得するステップである。

20

【0076】

先ず、本実施形態における前記情報取得ステップについて説明する。本実施形態においては、前記ライトプロテクト書き込み手段により書き込まれたファイルは、連続したWPブロックに書き込まれており、かつ該WPブロックの最終WPブロック直後のWPブロックはライトプロテクトされていないことが保証されている。そのため、該ファイルの位置を特定するためには、ライトプロテクトされているWPブロックを検索し、該WPブロックから連続してライトプロテクトされているWPブロック数を特定することで、該ファイルの開始位置及びサイズを特定することができる。

30

【0077】

続いて、本実施形態における前記ファイルデータ取得ステップについて説明する。前記情報取得ステップで取得したファイルデータの開始位置及びサイズにより、ファイルデータを取得することができる。

【0078】

図26は、該ライトプロテクトファイル復旧手段の動作をフローチャートで示している。図26を説明する。先ず、ステップS2601でSDメモ리카ード17のデータ領域の先頭から順に、ライトプロテクトされているWPブロックを検索し、ステップS2602で発見したかを判断する。発見した場合には、ステップS2603で連続して何個のWPブロックがライトプロテクトされているかを調べる。これにより、ライトプロテクトファイルの開始位置とサイズが確定する。

40

【0079】

続いて、ステップS2604で、ステップS2603で取得した連続WPブロック内のファイルデータを取得し、ステップS2605で該連続WPブロックのライトプロテクトを解除し、ステップS2602へ戻り繰り返す。ステップS2602で、発見しなかった場合は、正常終了する。

【0080】

以上説明した実施形態によれば、前記WPブロック境界一致手段、前記ライトプロテクト書き込み手段、及び前記ライトプロテクトファイル復旧手段によって、ファイルデータにライトプロテクトをかける際には、ファイルが格納されたクラスタのみをライトプロテ

50

クトすることにより、無駄のないファイルデータのライトプロテクトが行える。また、いずれのソフトウェア及び機器によってデータ管理がされても、ライトプロテクトしたファイルが削除されることはない。さらに、フォーマットなどにより、ファイルシステムが破壊されても、ライトプロテクトしたファイルを復旧することができる。

【0081】

(他の実施形態)

本実施形態においては、MBRを持たないようにフォーマットされた記憶装置を例に取り説明したが、MBRを持ち、パーティションを複数持つ記憶装置においても、パーティションの開始アドレスの取得手段があれば、本発明を適用できることは言うまでもない。

【0082】

さらに、本実施形態においては、記憶装置としてSDメモリカードを例に取り説明したが、いずれの記憶装置であっても、データ領域の部分的なライトプロテクト機能を持つ記憶装置においては、本発明を適用できることは言うまでもない。

【0083】

さらに、本実施形態においては、前記WPブロック境界一致手段により、クラスタとWPブロックの境界を一致させていることを前提に、前記ライトプロテクト書き込み手段、及びライトプロテクトファイル復旧手段を説明したが、クラスタとWPブロックの境界が一致していない場合であっても、プロテクトするファイルデータが一部分でも格納されるWPブロック全てにライトプロテクトをかける手段と、該手段によりライトプロテクトがかけられたWPブロックが一部分でも存在するクラスタを別のデータの書き込み対象となることを防止する手段とを具備する場合、本発明を適用できる。

【0084】

さらに、本実施形態においては、ライトプロテクトファイルの書き込み時、連続クラスタ領域に書き込むことを前提に説明したが、FAT領域をすべてライトプロテクトし、リンク情報をも保護しておけば、不連続なクラスタ領域にファイルを書き込んでも、本発明を適用できることは言うまでもない。

【0085】

さらに、本実施形態においては、初期状態として、前記ライトプロテクトディレクトリがルートディレクトリエントリの直後に位置し、かつ該クラスタの直後には使用済みの空きクラスタが位置し、かつライトプロテクトファイルの書き込み時には、該ライトプロテクトディレクトリに書き込むことを前提に説明したが、ライトプロテクトファイルを書き込むディレクトリエントリを全てライトプロテクトし、かつフォーマットされてしまった場合でも、クラスタサイズとクラスタ開始位置を特定する手段を具備する場合、本発明を適用できることは言うまでもない。

【0086】

さらに、本実施形態においては、FATファイルシステムを例にとり説明したが、記憶装置のデータ領域がいずれのファイルシステムにより管理されていても、クラスタ等のファイルの最小単位が存在する場合、本発明のWPブロック境界一致手段を適用し、本発明を適用できることは言うまでもない。

【図面の簡単な説明】

【0087】

【図1】本発明の第1の実施形態を示すファイル管理手段を適用可能な画像処理システムの一例を示すブロック図である。

【図2】図22に示したSDメモリカードを構成するファイルシステムの階層構造を示す概念図である。

【図3】第1の実施形態におけるライトプロテクト書き込み手段を示すフローチャート図である。

【図4】第1の実施形態における、本発明の第三の特徴であるライトプロテクトファイル復旧手段を示すフローチャート図である。

【図5】SDメモリカードの一般的な内部データ構造を示すブロック図である。

10

20

30

40

50

【図 6】図 5 における M B R の内部データ構造を示すブロック図である。

【図 7】図 5 におけるパーティションを F A T ファイルシステムでフォーマットした場合の内部データ構造を示すブロック図である。

【図 8】図 7 における P B R の内部データ構造を示すブロック図である。

【図 9】図 8 における B P B に含まれるパラメータを示すテーブルである。

【図 10】S D メモリカードのデータ領域の一例を示す状態図である。

【図 11】S D メモリカードの図 10 の状態におけるデータ構造を示す構造図である。

【図 12】S D メモリカードに対するファイル読み出し手段を示すフローチャート図である。

【図 13】S D メモリカードに対するファイル書き込み手段を示すフローチャート図である。

【図 14】S D メモリカードに対するファイル削除手段を示すフローチャート図である。

【図 15】S D メモリカードのライトプロテクト動作の手順を示すフローチャート図である。

【図 16】S D メモリカードのライトプロテクト解除動作の手順を示すフローチャート図である。

【図 17】S D メモリカードのライトプロテクト状態参照動作の手順を示すフローチャート図である。

【図 18】S D メモリカードに対するライトプロテクト書き込みを行う場合の、初期データ構造を示す図である。

【図 19】S D メモリカードにおける W P ブロック割り当て構造を示すデータ構造図である。

【図 20】実施形態 1 における S D メモリカードの固有パラメータの例を示す表である。

【図 21】図 20 のパラメータを持つ S D メモリカードに W P ブロック境界一致手段を適用した場合の B P B パラメータの例を示す表である。

【図 22】図 21 のパラメータを基にフォーマットされた S D メモリカードのデータ領域の構造を示す構造図である。

【図 23】F A T 1 6 におけるクラスタ開始アドレス計算式である。

【図 24】ライトプロテクト削除手段、及びライトプロテクト書き換え手段におけるライトプロテクト解除動作を示すフローチャート図である。

【図 25】第 2 の実施形態におけるライトプロテクト書き込み手段を示すフローチャート図である。

【図 26】第 2 の実施形態における、本発明の第三の特徴であるライトプロテクトファイル復旧手段を示すフローチャート図である。

【図 27】ライトプロテクトした場合における S D メモリカードのデータ領域の構造を示す構造図 1 である。

【図 28】ライトプロテクトした場合における S D メモリカードのデータ領域の構造を示す構造図 2 である。

【符号の説明】

【 0 0 8 8 】

- 1 レンズ
- 2 C C D ユニット
- 3 A / D コンバータ
- 4 S S G ユニット
- 5 C P U
- 6 信号処理アクセラレータ
- 7 電池
- 8 D C / D C コンバータ
- 9 電源コントローラ
- 1 0 サブ C P U

10

20

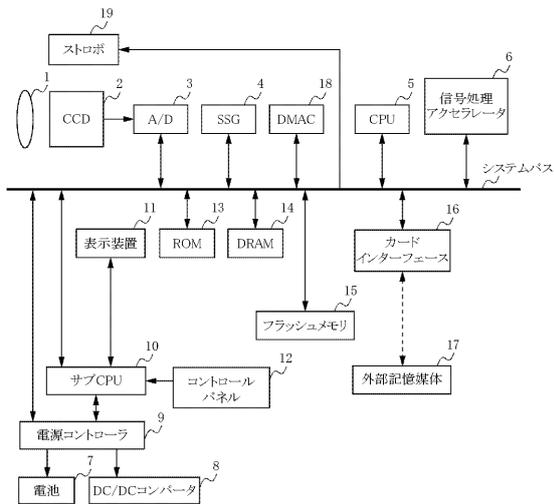
30

40

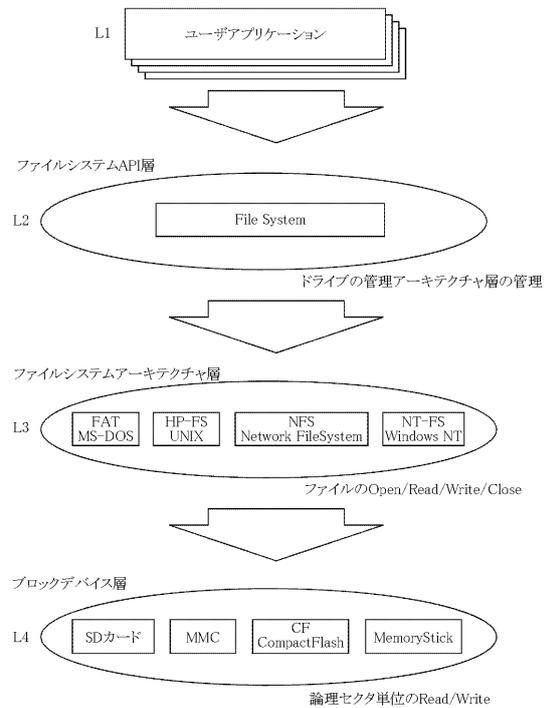
50

- 1 1 表示装置
- 1 2 コントロールパネル
- 1 3 R O M
- 1 4 D R A M
- 1 5 フラッシュ R O M
- 1 6 カードインターフェース
- 1 7 外部記憶媒体
- 1 8 D M A コントローラ
- 1 9 ストロボ

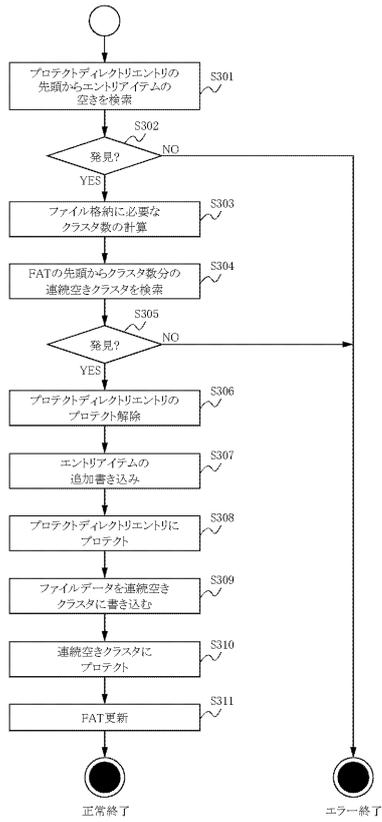
【 図 1 】



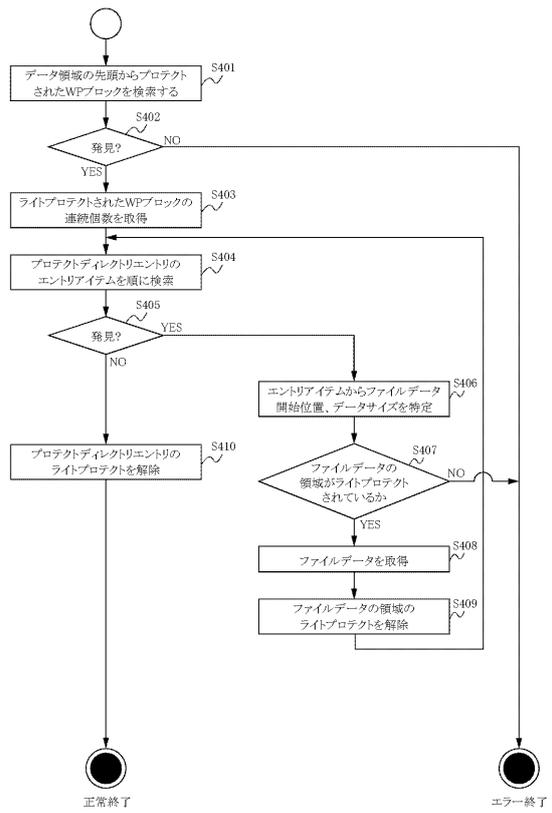
【 図 2 】



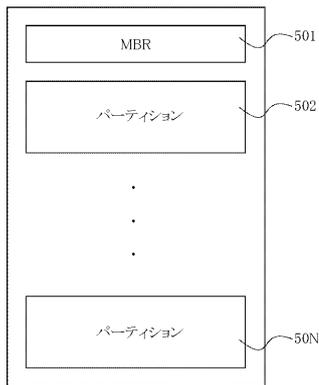
【図3】



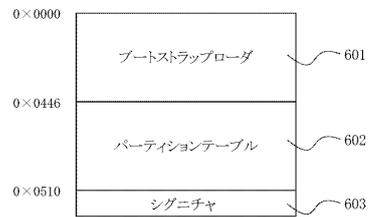
【図4】



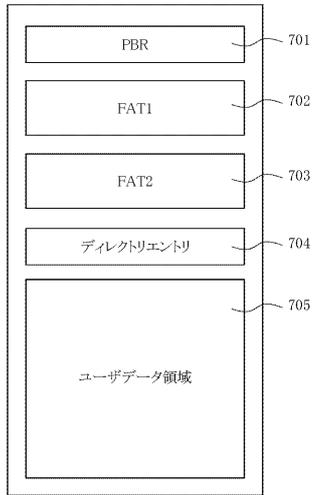
【図5】



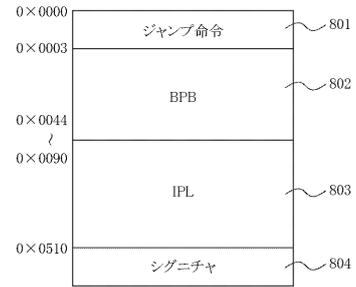
【図6】



【図7】



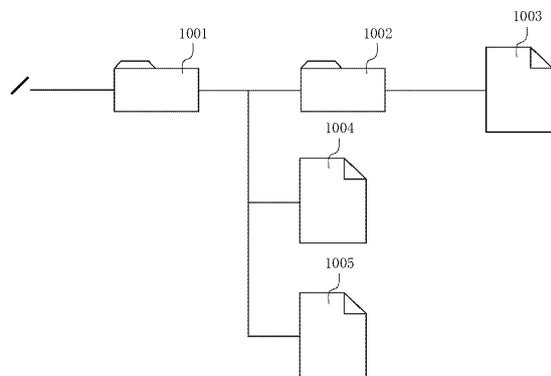
【図8】



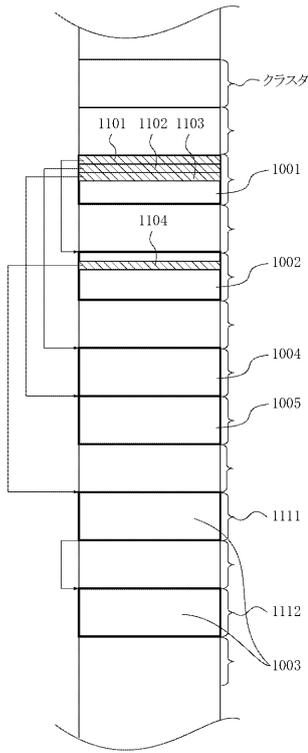
【図9】

サイズ(Byte)	内容	
8	OEM名	901
2	セクタサイズ	902
1	クラスターサイズ	903
2	リザーブセクタ数	904
1	FATの数	905
2	ルートディレクトリエントリ数	906
2	総セクタ数	907
1	メディアDescriptor	908
2	FATサイズ(セクタ単位)	909
2	トラックサイズ(セクタ単位)	910
2	ヘッド数	911
4	Hiddenセクタ数	912
4	32Bitセクタ数	913
1	ドライブID	914
1	Reserve	915
1	ブロードシグニチャ	916
4	シリアル番号	917
11	ボリューム名	918
8	ファイルシステムタイプ	919

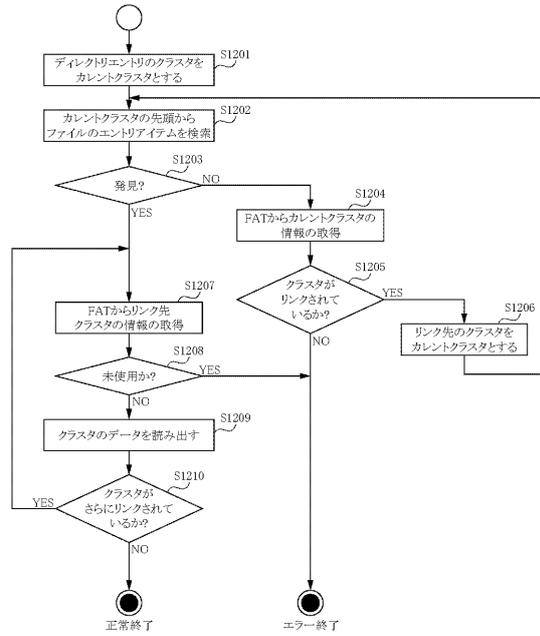
【図10】



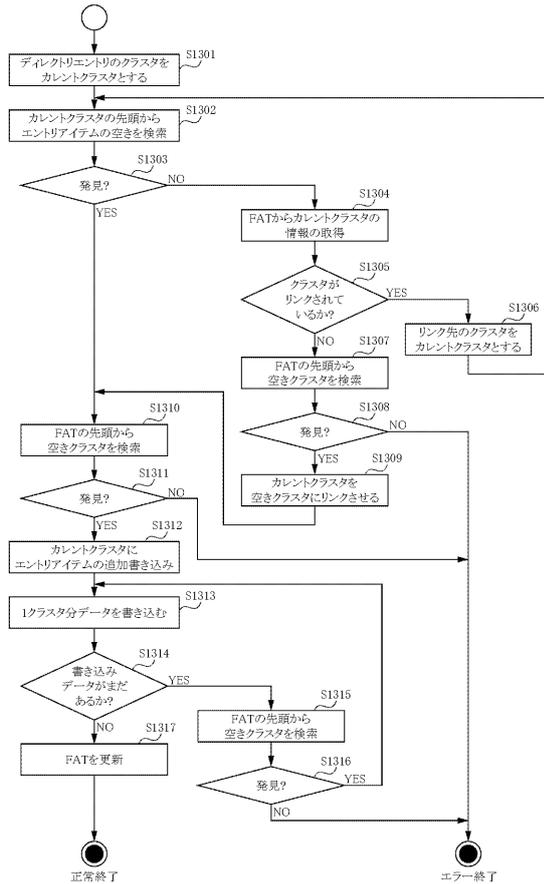
【図11】



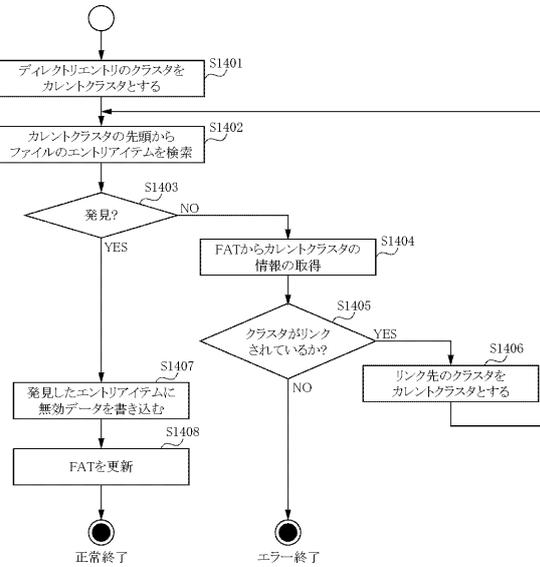
【図12】



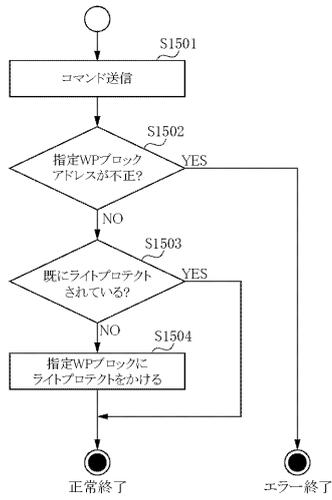
【図13】



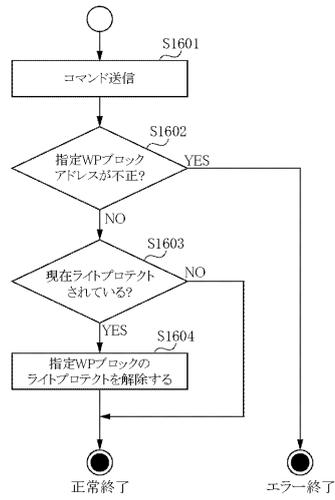
【図14】



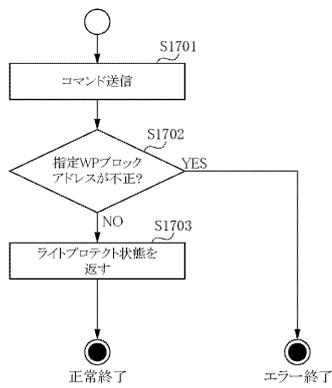
【図15】



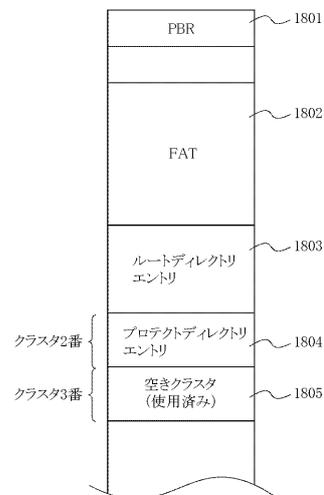
【図16】



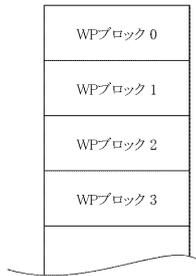
【図17】



【図18】



【図19】



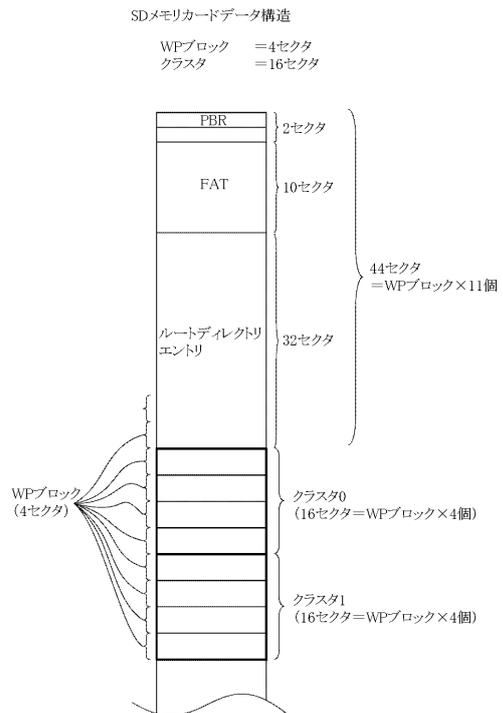
【図20】

パラメータ	値
総容量	10Mバイト(20480セクタ)
セクタサイズ	512バイト
WPブロックサイズ	2Kバイト

【図21】

パラメータ	値
セクタサイズ	512バイト
クラスタサイズ	8Kバイト
リザーブセクタ数	2セクタ
FATの数	2個
ルートディレクトリエントリ数	512個
総セクタ数	20480セクタ
FATサイズ	5セクタ

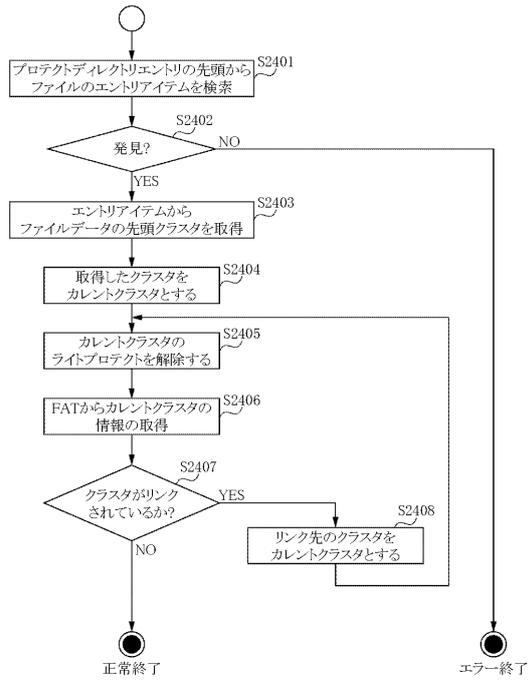
【図22】



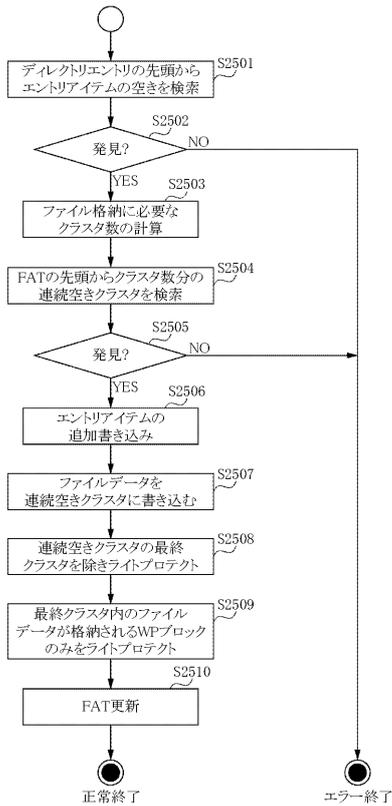
【図23】

$$\begin{aligned}
 & \text{クラスタ開始位置} \quad 2301 \\
 & = \text{パーティションの開始アドレス} \quad 2302 \quad 2304 \quad 2305 \quad 2303 \\
 & + (\text{リザーブセクタ数} + \text{FATサイズ} \times \text{FATの数}) \times \text{セクタサイズ} \\
 & + (\text{ルートディレクトリエントリ数} \times 32) \quad 2306
 \end{aligned}$$

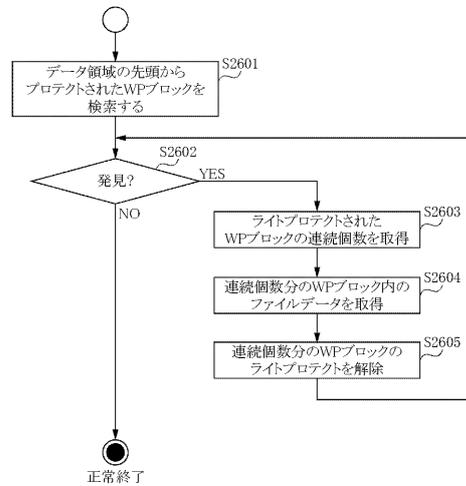
【図24】



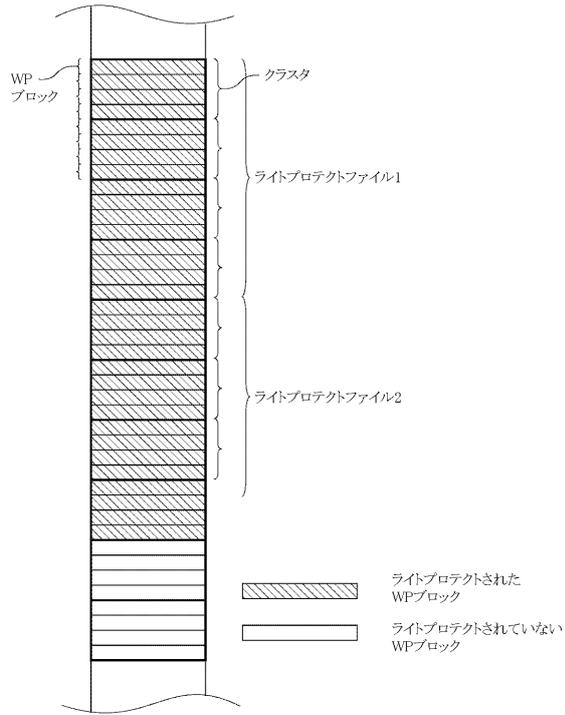
【図25】



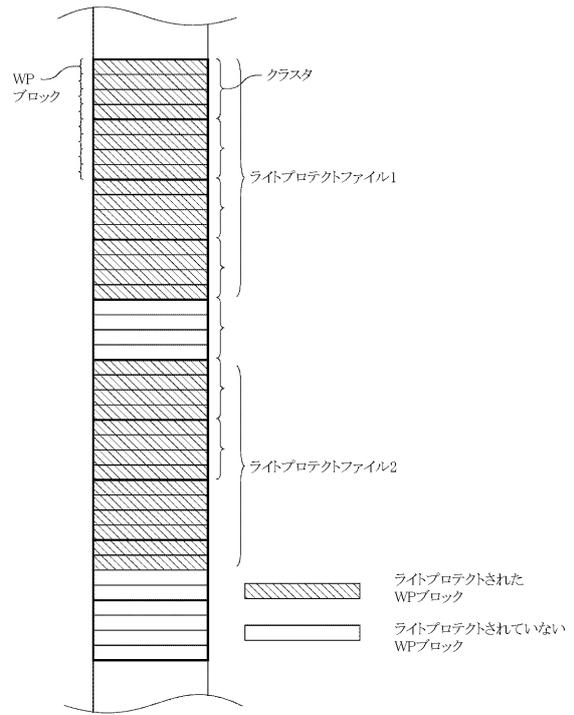
【図26】



【図 27】



【図 28】



フロントページの続き

(72)発明者 田川 陽次郎
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

審査官 滝谷 亮一

(56)参考文献 特開平02-193236(JP,A)
特開2001-043112(JP,A)
特開平10-124384(JP,A)
特開2003-085010(JP,A)
特開2001-188701(JP,A)

(58)調査した分野(Int.Cl., DB名)
G06F 12/14
G06F 21/02
G06F 21/24