

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4886866号  
(P4886866)

(45) 発行日 平成24年2月29日(2012.2.29)

(24) 登録日 平成23年12月16日(2011.12.16)

(51) Int.Cl.

F I

G 0 6 F 12/02 (2006.01)

G 0 6 F 12/02 5 1 0 A

請求項の数 4 (全 17 頁)

(21) 出願番号	特願2010-27558 (P2010-27558)	(73) 特許権者	390040187
(22) 出願日	平成22年2月10日 (2010.2.10)		株式会社バッファロー
(65) 公開番号	特開2011-164968 (P2011-164968A)		愛知県名古屋市中区大須三丁目30番20号
(43) 公開日	平成23年8月25日 (2011.8.25)	(74) 代理人	110000028
審査請求日	平成22年11月30日 (2010.11.30)		特許業務法人明成国際特許事務所
早期審査対象出願		(72) 発明者	福井 真吾
			名古屋市南区柴田本通4丁目15番 株式会社バッファロー内
		(72) 発明者	石井 俊
			名古屋市南区柴田本通4丁目15番 株式会社バッファロー内
		審査官	鈴木 和樹
			最終頁に続く

(54) 【発明の名称】 主記憶装置へのアクセスを高速化する方法および記憶装置システム

(57) 【特許請求の範囲】

【請求項 1】

コンピューターの主記憶装置の記憶領域の中の前記コンピューターのオペレーティングシステムが管理していない管理外記憶領域を用いて、前記主記憶装置へのアクセスを高速化する方法であって、

(a) 前記管理外記憶領域をRAMディスクとして利用する第1のソフトウェアが、前記RAMディスクの容量を指定するためのユーザーによる設定操作を受け付ける工程と、

(b) 前記コンピューターが再起動されたときに、前記工程(a)においてユーザーによって指定された前記RAMディスクの容量に応じて、前記主記憶装置における前記管理外記憶領域の範囲が変更されるように、前記第1のソフトウェアが設定する工程と、

(c) 前記第1のソフトウェアが、前記コンピューターの再起動後に、前記管理外記憶領域における連続した領域を、前記設定操作によって指定された容量を有するRAMディスクの記憶領域として設定する工程と、

を備える方法。

【請求項 2】

請求項1記載の方法であって、さらに、

(d) 前記第1のソフトウェアが、前記管理外記憶領域の中で、前記RAMディスクの記憶領域として使用していない未使用の記憶領域の範囲をレジストリに書き込む工程と、

(e) 第2のソフトウェアが、前記工程(c)において前記レジストリに書き込まれた情報に基づいて、前記未使用の記憶領域の範囲から優先的に前記コンピューターの外部記憶

10

20

装置のためのキャッシュ領域を確保する工程と、  
を備える、方法。

【請求項 3】

コンピューターに用いられる記憶装置システムであって、  
記憶領域の一部が前記コンピューターのオペレーティングシステムによって管理される  
主記憶装置と、

前記主記憶装置の記憶領域の中の前記オペレーティングシステムが管理していない管理  
外記憶領域の一部を R A M ディスクとして利用する第 1 のソフトウェアと、  
を備え、

前記第 1 のソフトウェアは、

前記 R A M ディスクの容量を指定するためのユーザーによる設定操作を受け付けるイン  
ターフェイスモジュールと、

前記コンピューターが再起動されたときに、前記インターフェイスモジュールが受け付  
けた前記 R A M ディスクの容量に応じて、前記主記憶装置における前記管理外記憶領域の  
範囲が変更されるように設定し、前記コンピューターの再起動後に、前記管理外記憶領域  
における連続した領域を、前記設定操作によって指定された容量を有する R A M ディスク  
の記憶領域として設定する記憶領域設定モジュールと、  
を備える、記憶装置システム。

【請求項 4】

請求項 3 記載の記憶装置システムであって、さらに、

前記主記憶装置の一部の記憶領域を前記コンピューターの外部記憶装置のためのキャッ  
シュ領域として使用する第 2 のソフトウェアを備え、

前記第 1 のソフトウェアは、前記管理外記憶領域の中で、前記 R A M ディスクの記憶領  
域として使用していない未使用の記憶領域の範囲をレジストリに書き込み、

前記第 2 のソフトウェアは、前記第 1 のソフトウェアが前記レジストリに書き込んだ情  
報に基づいて、前記未使用の記憶領域の範囲から優先的に前記外部記憶装置のためのキャ  
ッシュ領域を確保する、記憶装置システム。

【発明の詳細な説明】

【技術分野】

【0001】

この発明は、コンピューターの主記憶装置へのアクセスに関する。

【背景技術】

【0002】

コンピューターは、通常、オペレーティングシステム (Operating System ; O S ) によ  
って、その主記憶装置における記憶領域が管理される。O S は、コンピューターで実行さ  
れる各プログラムからの要求があったときに、主記憶装置の記憶領域における空き領域か  
ら、その要求に応じた容量の領域を割り当てる。そして、O S は、記憶領域の割り当てを  
受けたプログラムがその領域の使用を終了したときに、当該領域を開放する。この記憶領  
域の割り当てと開放の繰り返しにより、主記憶装置の記憶領域は、通常、各プログラムの  
割り当て領域ごとに細かく分断された状態となる。

【0003】

ここで、主記憶装置の記憶領域の一部を仮想的な外部記憶装置として使用する「R A M  
ディスク」と呼ばれる技術が知られている (下記特許文献 1 等)。主記憶装置において、  
この R A M ディスクを設定する場合に、上記のように、主記憶装置の記憶領域が細かく分  
断されていると、R A M ディスクとして割り当てられる記憶領域も細かく分断されてしま  
う。すると、ユーザーが R A M ディスクにアクセスする際のアクセス速度が低下してしま  
うという問題があった。これまで、こうした問題に対して十分な工夫がなされてこなかっ  
たのが実情であった。

【先行技術文献】

【特許文献】

【 0 0 0 4 】

【特許文献 1】特開平 1 1 - 8 5 6 0 7 号公報

【発明の概要】

【発明が解決しようとする課題】

【 0 0 0 5 】

本発明は、コンピュータに用いられる主記憶装置へのアクセスを高速化する技術を提供することを目的とする。

【課題を解決するための手段】

【 0 0 0 6 】

本発明は、上述の課題の少なくとも一部を解決するためになされたものであり、以下の形態又は適用例として実現することが可能である。

10

【 0 0 0 7 】

[ 適用例 1 ]

コンピュータの主記憶装置の記憶領域の中の前記コンピュータのオペレーティングシステムが管理していない管理外記憶領域を用いて、前記主記憶装置へのアクセスを高速化する方法であって、

( a ) 前記主記憶装置の一部の領域を R A M ディスクとして利用する第 1 のソフトウェアが、前記 R A M ディスクの容量を指定するためのユーザーによる設定操作を受け付ける工程と、

( b ) 前記第 1 のソフトウェアが、前記コンピュータの起動時に、前記管理外記憶領域における連続した領域を、前記設定操作によって指定された容量を有する R A M ディスクの記憶領域として設定する工程と、

20

を備え、

前記工程 ( b ) は、前記第 1 のソフトウェアが、前記工程 ( a ) においてユーザーによって指定された前記 R A M ディスクの容量に応じて、前記オペレーティングシステムの管理する前記主記憶装置の記憶領域の範囲を変更して、前記 R A M ディスクのための記憶領域を確保する工程を含む、方法。

この方法によれば、ユーザーが指定した容量を有する R A M ディスクの記憶領域を、主記憶装置の管理外記憶領域における連続した領域に設定することができる。そのため、主記憶装置上の R A M ディスクへのアクセス速度を向上させることができる。

30

【 0 0 0 8 】

[ 適用例 2 ]

適用例 1 記載の方法であって、さらに、

( c ) 前記第 1 のソフトウェアが、前記管理外記憶領域の中で、前記 R A M ディスクの記憶領域として使用していない未使用の記憶領域の範囲をレジストリに書き込む工程と、

( d ) 第 2 のソフトウェアが、前記工程 ( c ) において前記レジストリに書き込まれた情報に基づいて、前記未使用の記憶領域の範囲から優先的に前記コンピュータの外部記憶装置のためのキャッシュ領域を確保する工程と、

を備える、方法。

この方法によれば、管理外記憶領域における連続した領域を R A M ディスクやソリッドステートドライブなどの外部記憶装置のためのキャッシュ領域として利用することができる。従って、R A M ディスクへのアクセスを高速化するとともに、ユーザーが体感する外部記憶装置へのアクセス速度を向上させることができる。また、この方法では、レジストリを、第 1 のソフトウェアから第 2 のソフトウェアに管理外記憶領域の空き領域を通知するための手段として利用する。従って、オペレーティングシステムによって管理されていない管理外記憶領域を異なるプログラム同士が重複して使用してしまうことによるエラーの発生を回避することができる。

40

【 0 0 0 9 】

[ 適用例 3 ]

コンピュータに用いられる記憶装置システムであって、

50

記憶領域の一部が前記コンピュータのオペレーティングシステムによって管理される主記憶装置と、

前記主記憶装置の記憶領域の中の前記オペレーティングシステムが管理していない管理外記憶領域の一部をＲＡＭディスクとして利用する第１のソフトウェアと、

を備え、

前記第１のソフトウェアは、

前記ＲＡＭディスクの容量を指定するためのユーザーによる設定操作を受け付けるインターフェイスモジュールと、

前記コンピュータの起動時に、前記管理外記憶領域における連続した領域を、前記設定操作によって指定された容量を有するＲＡＭディスクの記憶領域として設定する記憶領域設定モジュールと、

を備え、

前記記憶領域設定モジュールは、前記インターフェイスモジュールが受け付けた、ユーザーの指定した前記ＲＡＭディスクの容量に応じて、前記オペレーティングシステムの管理する前記主記憶装置の記憶領域の範囲を変更して、前記ＲＡＭディスクのための記憶領域を確保する、記憶装置システム。

【００１０】

[適用例４]

適用例３記載の記憶装置システムであって、さらに、

前記主記憶装置の一部の記憶領域を前記コンピュータの外部記憶装置のためのキャッシュ領域として使用する第２のソフトウェアを備え、

前記第１のソフトウェアは、前記管理外記憶領域の中で、前記ＲＡＭディスクの記憶領域として使用していない未使用の記憶領域の範囲をレジストリに書き込み、

前記第２のソフトウェアは、前記第１のソフトウェアが前記レジストリに書き込んだ情報に基づいて、前記未使用の記憶領域の範囲から優先的に前記外部記憶装置のためのキャッシュ領域を確保する、記憶装置システム。

【００１１】

なお、本発明は、種々の形態で実現することが可能であり、例えば、主記憶装置へのアクセスを高速化する方法およびその方法を実行する記憶装置システム、それらの方法またはシステムの機能を実現するためのコンピュータプログラム、そのコンピュータプログラムを記録した記録媒体等の形態で実現することができる。

【図面の簡単な説明】

【００１２】

【図１】第１実施例におけるコンピュータの構成を示す概略図。

【図２】ＲＡＭの記憶領域と、本実施例のコンピュータにおけるアドレス空間の割り当てを説明するための模式図。

【図３】ＲＡＭディスクドライバがＲＡＭディスクを設定するために実行する処理の処理手順を示すフローチャート。

【図４】ＲＡＭディスクの容量を設定するための操作画面を示す模式図。

【図５】コンピュータの再起動後におけるアドレス空間を示す模式図。

【図６】第１と第２の比較例を説明するためのアドレス空間を示す模式図。

【図７】第２実施例におけるコンピュータの構成を示す概略図。

【図８】ＲＡＭディスクドライバがＲＡＭディスクを設定するために実行する処理の処理手順を示すフローチャート。

【図９】レジストリ設定の一例を示す説明図。

【図１０】ＳＳＤドライバがＲＡＭ上にＳＳＤ用キャッシュ領域を設定するために実行する処理の処理手順を示すフローチャート。

【図１１】ＲＡＭ上にＲＡＭディスクとＳＳＤ用キャッシュ領域とが設定された後のアドレス空間を示す模式図。

【発明を実施するための形態】

10

20

30

40

50

## 【 0 0 1 3 】

## A. 第 1 実施例：

図 1 は本発明の一実施例としてのコンピューターの構成を示す概略図である。コンピューター 100 は、中央処理装置 (Central Processing Unit ; CPU) 110 と、主記憶装置としての RAM (Random Access Memory) 120 と、外部記憶装置 130 と、表示部 140 と、操作部 150 とを備える。これらの各構成部 110 ~ 150 は、内部バス 101 を介して互いに接続されている。

## 【 0 0 1 4 】

CPU 110 は、外部記憶装置 130 に格納された OS やプログラム (図示せず) を読み出し、RAM 120 上に展開して実行する。図 1 は、CPU 110 がデバイスドライバの一種である RAM ディスクドライバ 112 を実行している状態を示している。RAM ディスクドライバ 112 の機能については後述する。なお、本実施例のコンピューター 100 では、OS として Windows XP (「WINDOWS XP」はマイクロソフト コーポレーションの登録商標) などの 32 ビット OS が導入されているものとする。

## 【 0 0 1 5 】

RAM 120 は、揮発性の記憶素子であり、その記憶領域の割り当てや開放は OS によって管理されている。なお、図 1 は、RAM 120 の記憶領域の一部に RAM ディスク 122 が設定されている状態を示している。RAM ディスク 122 については後述する。外部記憶装置 130 は、ハードディスクドライブ (Hard disk drive ; HDD) によって構成された不揮発性の記憶装置であり、OS やプログラムの他、各プログラムで利用される設定データなどが格納されている。表示部 140 は、液晶ディスプレイなどのディスプレイ装置によって構成される。操作部 150 は、キーボードやマウスなどの入力デバイスによって構成され、ユーザーの操作を受け付ける。

## 【 0 0 1 6 】

図 2 は、RAM 120 における記憶領域 (物理メモリ) と、コンピューター 100 におけるアドレス空間の割り当てを説明するための模式図である。本実施例では、RAM 120 は、5.0 ギガバイト (GB) の記憶容量を有しているものとする。一般に、32 ビット OS は、4.0 GB までのアドレス空間を設定することができる。ここで、4.0 GB のアドレス空間中の末尾の領域には、I/O デバイスやシステムの制御のために用いられる固定的な領域が確保される。本明細書では、この領域を「制御用領域」と呼ぶ。制御用領域としては、例えば、0.5 GB 分の領域が確保される。

## 【 0 0 1 7 】

そして、アドレス空間のうち、先頭アドレスから制御用領域までのアドレス空間が、物理メモリを管理するために割り当てられる。即ち、32 ビット OS では、RAM 120 が 5.0 GB の容量を有していても、制御用領域として 0.5 GB 分のアドレス空間が確保される場合には、RAM 120 の全記憶領域のうち、先頭から最大 3.5 GB 分の記憶領域しか管理することができない。本明細書では、OS によって管理される領域を「OS 管理領域」と呼び、OS による管理が及ばない領域を「OS 管理外領域」と呼ぶ。RAM 120 上の OS 管理領域には、OS やデバイスドライバを含む各種のプログラム、それらのプログラムにおいて用いられる各種のデータが格納される。また、RAM 120 上の OS 管理領域に格納された各種のデータには、OS やプログラムが直接的にアクセスすることが可能である。

## 【 0 0 1 8 】

ここで、本実施例のコンピューター 100 では、RAM ディスクドライバ 112 が、制御用領域に後続するアドレス空間を設定し、RAM 120 の OS 管理外領域に割り当てる。図 2 には、制御用領域に続けて 4.0 ~ 5.5 GB の範囲のアドレス空間 (破線で図示) が設定され、RAM 120 の 1.5 GB 分の OS 管理外領域に割り当てられている状態が示されている。これによって、RAM ディスクドライバ 112 は、OS 管理外領域へのアクセスを可能とし、その上で、OS 管理外領域に仮想的な外部記憶装置である RA

Mディスク122の記憶領域を設定する。

【0019】

図3は、RAMディスクドライバー112が、RAMディスク122を設定するために実行する処理の処理手順を示すフローチャートである。ステップS10では、RAMディスクドライバー112は、コンピューター100の物理メモリの容量(RAM120の容量)と、RAM120におけるOS管理領域の範囲とを取得する。具体的には、RAMディスクドライバー112は、これらの情報を、BIOS(Basic Input Output System)などのデバイス制御のためのプログラムから取得するものとしても良い。ステップS20では、RAMディスクドライバー112は、ユーザーによるRAMディスク122の容量を指定するための設定操作を受け付ける。

10

【0020】

図4は、RAMディスクドライバー112がコンピューター100の表示部140に表示させる操作画面を示す模式図である。コンピューター100のユーザーはこの操作画面10に対する画像上の操作によって、RAMディスク122の容量を指定することができる。操作画面10は、第1と第2のレベルバー11、12と、第1と第2のチェックボックス11c、12cと、スライダー13と、決定ボタン14と、終了ボタン15とを有する。

【0021】

ここで、図4(A)は、第1のレベルバー11に対する操作が実行されている状態を示しており、図4(B)は、第2のレベルバー12に対する操作が実行されている状態を示している。第1と第2のレベルバー11、12はそれぞれ、RAMディスク122の容量を示す帯グラフ状の画像オブジェクトである。本実施例では、RAMディスク122の容量としてユーザーは0GBから3.0GBまでの容量を指定できる。具体的には、ユーザーは、RAM120のOS管理外領域の容量に相当する範囲(図では0~1.5GBの範囲)で、RAMディスク122の容量を指定するときは、第1のレベルバー11に対して操作を実行することができる。一方、OS管理外領域の容量を超えて、RAMディスク122の容量を指定するときは、第2のレベルバー12に対して操作を実行することができる。

20

【0022】

なお、第1のレベルバー11に対してRAMディスク122の容量を指定する操作が実行されたときには、RAMディスクドライバー112は、後述する処理において、OS管理領域の範囲を変更することなく、RAMディスク122を設定することとなる。また、第2のレベルバー12に対してRAMディスク122の容量を指定する操作が実行されたときには、RAMディスクドライバー112は、指定された容量に応じて、RAM120におけるOS管理領域の範囲を縮小させる処理を実行することとなる。

30

【0023】

操作画面10における具体的な設定操作は以下の通りである。ユーザーは、第1と第2のレベルバー11、12のそれぞれに対応して設けられた第1と第2のチェックボックス11c、12cのいずれかにチェックマーク(図では黒丸で表示)を入れることによって操作対象を選択する。そして、ユーザーは、第1と第2のレベルバー11、12のそれぞれに対応して設けられたスライダー13のうち操作対象となる方(図では黒く塗りつぶされている方)を、レベルバー11、12上の所望の容量を示す位置まで移動させ、決定ボタン14を押下する。これによって、RAMディスク122の容量が指定される。

40

【0024】

なお、この設定操作をキャンセルする場合には、ユーザーは、決定ボタン14を押下することなく、終了ボタン15を押下すれば良い。また、決定ボタン14を押下した後に、ユーザーに設定の確認を促す旨のメッセージが表示されるものとしても良く、このときに、ユーザーは、設定操作を承認するか、無効とするかを選択できるものとしても良い。

【0025】

ステップS30では、RAMディスクドライバー112は、コンピューター100にR

50

RAMディスク122が設定されるための処理を実行する。具体的には、RAMディスクドライバ112は、コンピューター100の再起動後に、RAM120のOS管理外領域に対してアドレス空間が割り当てられ、RAMディスクドライバ112がCPU110にロードされるように、起動ファイルを編集する。

#### 【0026】

ここで、ステップS20において、ユーザーがRAMディスク122の容量として、OS管理領域の範囲を越える容量を指定したときには、RAMディスクドライバ112は、OS管理領域の範囲を縮小させるために、以下の処理を実行する。即ち、RAMディスクドライバ112は、RAM120の容量とユーザーが指定したRAMディスク122の容量との差を取り、縮小後のOS管理領域の範囲を算出する。そして、起動ファイルに含まれるOS用のBoot.iniファイルに、OSが使用できるメモリの最大容量を設定するための/MAXMEM=nnスイッチを配置し、縮小後のOS管理領域の範囲を指定する。ここで、/MAXMEM=nnスイッチは、OS管理領域として設定する物理メモリの容量をメガバイト(MB)単位で指定することができる。具体的には、例えば、OS管理領域の範囲を0~2.0GBの範囲とする場合には、Boot.iniファイルに、「/MAXMEM=2048」と記述する。

#### 【0027】

ステップS40では、RAMディスクドライバ112は、表示部140に、コンピューター100の再起動を促す旨のメッセージを表示させ、ユーザーにコンピューター100の再起動を実行させる。コンピューター100の再起動により、ステップS30において編集された起動ファイルがCPU110によって読み込まれる。これによって、RAM120のOS管理外領域にアドレス空間が割り当てられる。また、RAMディスクドライバ112が起動され、RAM120のOS管理外領域にRAMディスク122の記憶領域が設定される。

#### 【0028】

図5(A)~(C)はそれぞれ、ステップS40におけるコンピューター100の再起動後のアドレス空間を示す模式図である。なお、図5(A)~(C)では、便宜上、制御用領域が破線で図示されている。

#### 【0029】

図5(A)は、ステップS20においてユーザーがRAMディスク122の容量としてOS管理外領域の容量より小さい容量を指定したときのアドレス空間を示している。RAMディスクドライバ112は、RAM120においてOS管理領域として割り当てられるメモリ範囲(図では0~3.5GBの範囲)を変更することなく、ユーザーが指定した容量で、アドレス空間の末尾からの連続領域にRAMディスク122の記憶領域を確保する。この場合には、RAM120のOS管理外領域において未使用の記憶領域が生じる。

#### 【0030】

図5(B)は、ステップS20においてユーザーがRAMディスク122の容量として、OS管理外領域の容量と同じ容量を指定したときのアドレス空間を示している。この場合には、RAM120のOS管理外領域に割り当てられるメモリ範囲の全範囲がRAMディスク122の記憶領域として確保される。図5(C)は、ステップS20においてユーザーがRAMディスク122の容量として、RAM120におけるOS管理外領域の容量より大きい容量を指定した場合のアドレス空間を示している。この場合には、RAMディスクドライバ112は、RAM120におけるOS管理領域の範囲を縮小させ、制御用領域の両側の2つの連続領域をRAM120のOS管理外領域に割り当てる。そして、RAM120のOS管理外領域に、ユーザーが指定した容量の分だけRAMディスク122のための記憶領域を確保する。なお、図5(C)では、RAM120におけるOS管理領域の範囲が0~2.0GBの範囲まで縮小され、拡張されたRAM120のOS管理外領域の全範囲にRAMディスク122が設定された状態が図示されている。

#### 【0031】

図6(A)、(B)はそれぞれ、第1と第2の比較例を説明するためのアドレス空間を

示す模式図である。図6(A)は、第1の比較例として、RAMディスク122がRAM120のOS管理領域内のみに設定される場合のアドレス空間を示している。ここで、RAM120上のOS管理領域内における記憶領域の割り当ては、OSが動的に実行する。即ち、OSは、各プログラムからの要求に応じて、必要な記憶領域の確保と開放とを繰り返す。そのため、この場合には、RAMディスク122の記憶領域は分断された状態で設定されてしまう。このように、RAMディスク122の記憶領域が分断されていると、RAMディスク122に対するアクセス速度は低下してしまう。また、第1の比較例の場合には、RAM120におけるOS管理外領域が未使用の記憶領域となってしまう。

【0032】

図6(B)は、第2の比較例として、RAMディスク122の記憶領域がRAM120のOS管理外領域とOS管理領域とに割り当てられる場合のアドレス空間を示している。第2の比較例では、ユーザーがRAM120におけるOS管理外領域の容量以上の容量をRAMディスク122の容量として指定した場合であっても、RAM120におけるOS管理領域の範囲が変更されない。この場合には、RAMディスク122のための記憶領域は、RAM120上においてOS管理外領域とOS管理領域の2つの領域に渡って割り当てられる。ただし、OS管理外領域におけるRAMディスク122の記憶領域は、連続した記憶領域として確保されるが、OS管理領域に含まれるRAMディスク122の記憶領域は分断された不連続な記憶領域として割り当てられてしまう。従って、この第2の比較例の場合にも、RAMディスク122に対するアクセス速度が低下する。

【0033】

このように、本実施例のコンピューター100であれば、RAMディスクドライバー112が、ユーザーが指定したRAMディスク122の容量に応じて、RAM120におけるOS管理領域の範囲を変更する。そして、RAMディスク122の記憶領域として、OS管理外領域における連続した記憶領域を確保する。従って、RAMディスク122へのアクセス速度を向上させることができる。また、RAM120のOS管理外領域を有効利用することが可能である。

【0034】

B. 第2実施例：

図7は、本発明の第2実施例としてのコンピューターの構成を示す概略図である。図7は、CPU110においてSSDドライバー114が実行されている点と、RAM120にSSD用キャッシュ領域124が追加されている点と、外部記憶装置130Aがソリッドステートドライブ(Solid State Drive; SSD)によって構成されている点以外は、図1とほぼ同じである。

【0035】

SSDドライバー114は外部記憶装置130A(以後、「SSD130A」とも呼ぶ)を制御するためのデバイスドライバーである。ここで、SSD130Aは、複数のフラッシュメモリが並列に接続された記憶装置である。フラッシュメモリの記憶領域はブロック単位で構成されており、フラッシュメモリへのデータの書き込み・消去もブロック単位で実行される。また、フラッシュメモリでは、データの更新は、既書き込まれている旧データを一旦消去した上で、更新データを新たに書き込むことにより実行される。即ち、SSDでは、微小なサイズのデータの更新処理の場合であっても、ブロック単位でのデータの消去・書き込みが実行されるため、データの更新処理においては、データ量に対する更新処理時間の割合が大きくなる傾向にある。従って、ユーザーは、SSD130Aへのアクセス速度が低下していると感得してしまう場合がある。

【0036】

そこで、第2実施例のコンピューター100Aでは、SSDドライバー114が、RAM120のOS管理外領域にSSD用キャッシュ領域124を設けることにより、SSD130Aに対するアクセスを効率的に実行する。これによって、ユーザーはあたかも、SSD130Aに対するアクセス速度が向上しているように感じることができる。具体的な、SSDドライバー114によるSSD130Aに対するアクセス処理については後述す



る。

#### 【 0 0 3 7 】

ところで、コンピューター 1 0 0 A では、第 1 実施例のコンピューター 1 0 0 と同様に、RAM ディスクドライバ 1 1 2 が、RAM 1 2 0 の OS 管理外領域に RAM ディスク 1 2 2 を設定する。即ち、コンピューター 1 0 0 A では、RAM 1 2 0 の OS 管理外領域に、2 つの異なるプログラムによって、2 つの記憶領域が設定されることとなる。しかし、OS 管理外領域には OS の管理が及ばないため、2 つのプログラムが別個に、互いに重複した記憶領域を使用してしまう可能性がある。そこで、第 2 実施例のコンピューター 1 0 0 A では、以下の処理手順により、RAM 1 2 0 の OS 管理外領域において RAM ディスクドライバ 1 1 2 と SSD ドライバ 1 1 4 とが別個に記憶領域を重複することなく

10

#### 【 0 0 3 8 】

図 8 は、RAM ディスクドライバ 1 1 2 が、RAM 1 2 0 に RAM ディスク 1 2 2 を設定するために実行する処理の処理手順を示すフローチャートである。図 8 は、ステップ S 3 5 が追加されている点以外は、図 3 とほぼ同じである。ステップ S 1 0 ~ ステップ S 3 0 では、第 1 実施例で説明したのと同様な処理が実行される。即ち、RAM ディスクドライバ 1 1 2 は、ユーザーからの RAM ディスク 1 2 2 の容量を指定する設定操作を受け付け、コンピューター 1 0 0 A の再起動後に、指定された容量の RAM ディスク 1 2 2 が、RAM 1 2 0 の OS 管理外領域に設定されるようにする。

#### 【 0 0 3 9 】

20

ステップ S 3 5 では、RAM ディスクドライバ 1 1 2 は、RAM ディスク 1 2 2 として使用しない RAM 1 2 0 上の OS 管理外領域の空き領域に関する情報をレジストリに書き込む。ここで、「レジストリ」とは、Windows 系 OS において、システムやアプリケーションソフトの設定データを登録するためのデータベースを意味する。コンピューター 1 0 0 A では、レジストリは SSD 1 3 0 A に格納されている。また、「RAM 1 2 0 上の OS 管理外領域の空き領域に関する情報」としては、RAM ディスク 1 2 2 の記憶領域として割り当てられていない OS 管理外領域の範囲を示すアドレス情報や、当該範囲の容量などがある。

#### 【 0 0 4 0 】

図 9 は、RAM ディスクドライバ 1 1 2 がレジストリに格納する設定内容の一例を示す説明図である。RAM ディスクドライバ 1 1 2 は、レジストリに名称を「Enable Over 4 G」として「1」または「0」の数値を書き込む。ここで、「REG\_DWORD」は数値型のデータであることを意味する。RAM ディスクドライバ 1 1 2 は、RAM 1 2 0 の OS 管理外領域に空きがなく、OS 管理外領域が使用できない状態であるときには、「Enable Over 4 G」に「0」を書き込む。また、RAM ディスクドライバ 1 1 2 は、RAM 1 2 0 の OS 管理外領域に空きがあり、OS 管理外領域が使用できる状態である場合には、「Enable Over 4 G」に「1」を書き込む。

30

#### 【 0 0 4 1 】

RAM ディスクドライバ 1 1 2 は、名称を「Start Addr」として、OS 管理外領域における空き領域の先頭アドレスをレジストリに格納する。即ち、「Start Addr」には、SSD 用キャッシュ領域 1 2 4 として使用可能な領域の先頭物理アドレスが格納される。さらに、RAM ディスクドライバ 1 1 2 は、名称を「Cache Size」として、RAM 1 2 0 上の OS 管理外領域における空き領域の容量をレジストリに格納する。即ち、「Cache Size」には、RAM 1 2 0 の OS 管理外領域において SSD 用キャッシュ領域 1 2 4 として使用可能な容量が格納される。

40

#### 【 0 0 4 2 】

ステップ S 4 0 では、RAM ディスクドライバ 1 1 2 が、コンピューター 1 0 0 A の再起動を促すメッセージを表示部 1 4 0 に表示させ、ユーザーにコンピューター 1 0 0 A を再起動させる。なお、再起動後のコンピューター 1 0 0 A では、RAM ディスクドライバ 1 1 2 がロードされ、起動ファイルにおける設定に従って、RAM 1 2 0 の OS 管理

50

外領域にRAMディスク122が設定される。

【0043】

図10は、SSDドライバー114がRAM120上にSSD用キャッシュ領域124を設定するために実行する処理の処理手順を示すフローチャートである。この処理は、図8のステップS40において再起動されたコンピューター100Aにおいて実行される処理である。なお、SSDドライバー114は、この再起動の際に、CPU110によって自動的にロードされるように予め起動ファイルに設定されている。この起動ファイルの設定は、再起動前のコンピューター100Aにおいて、SSDドライバー114自身によってなされるものとしても良い。

【0044】

ステップS110では、SSDドライバー114は、RAMディスクドライバー112が図8のステップS35においてレジストリに書き込んだ情報を読み出す。ステップS120では、SSDドライバー114は、レジストリから読み込んだ情報と、設定すべきSSD用キャッシュ領域124の容量とに基づいて、RAM120上に、SSD用キャッシュ領域124のための記憶領域を確保する。なお、「SSD用キャッシュ領域124の容量」は、再起動前のコンピューター100Aにおいて、予めユーザーが指定した容量であるものとしても良いし、予めSSDドライバー114が規定値として有している容量であるものとしても良い。

【0045】

ここで、ステップS20において、SSDドライバー114が図9で説明したレジストリの設定内容を読み込んだ場合のステップS30における処理を説明する。レジストリに格納された「EnableOver4G」の値が「0」のときには、RAM120のOS管理外領域にSSD用キャッシュ領域124を設定できない。そのため、SSDドライバー114は、OSに依頼してRAM120のOS管理領域内にSSD用キャッシュ領域124のための記憶領域を確保する。

【0046】

一方、「EnableOver4G」の値が「1」のときには、SSDドライバー114は、RAM120のOS管理外領域から優先的にSSD用キャッシュ領域124の記憶領域を確保する。即ち、SSDドライバー114は、レジストリから「StartAddr」に格納されたアドレスを読み出し、当該アドレスを先頭アドレスとして、SSD用キャッシュ領域124の容量に応じた連続する記憶領域を確保する。そして、設定すべきSSD用キャッシュ領域124の容量がレジストリの「CacheSize」に格納されている容量の値より大きいときには、SSDドライバー114は、RAM120のOS管理外領域における空き領域をSSD用キャッシュ領域124の記憶領域として割り当てた上で、不足する容量を、OSに依頼して、RAM120のOS管理領域から割り当てる。

【0047】

図11(A)~(C)はそれぞれ、RAM120上にRAMディスク122とSSD用キャッシュ領域124とが設定された後のアドレス空間を示す模式図である。図11(A)は、RAM120のOS管理外領域に、RAMディスク122の記憶領域とSSD用キャッシュ領域124とが割り当てられた状態を示している。RAMディスク122の記憶領域は、第1実施例でも説明したように、RAM120のOS管理外領域の末尾からユーザーが指定した容量の範囲の連続領域が割り当てられる。そして、SSD用キャッシュ領域124は、レジストリに書き込まれた情報に基づき、OS管理外領域において、RAMディスク122が使用していない記憶領域に設定される。

【0048】

なお、図11(A)では、SSD用キャッシュ領域124は、RAM120のOS管理外領域においてRAMディスク122が使用していない領域を全範囲として設定されている。しかし、RAMディスク122の容量とSSD用キャッシュ領域124の容量との和がRAM120におけるOS管理外領域の容量より小さい場合には、RAM120のOS管理外領域には、未使用の記憶領域が形成される。

10

20

30

40

50

## 【 0 0 4 9 】

図 1 1 ( B ) は、S S D 用キャッシュ領域 1 2 4 の容量が R A M 1 2 0 の O S 管理外領域だけでは賄えなかった場合のアドレス空間を示している。この場合には、前記したとおり、S S D ドライバー 1 1 4 は、R A M 1 2 0 の O S 管理外領域では賄えない容量を、O S に依頼して、R A M 1 2 0 の O S 管理領域において確保する。なお、R A M 1 2 0 の O S 管理領域からは、S S D 1 3 0 A におけるデータ管理の単位であるブロックの容量（例えば 6 4 k B ）又はその 1 / 2 のサイズに相当する固定長の容量を一単位として、S S D 用キャッシュ領域 1 2 4 の記憶領域に割り当てられることが好ましい。

## 【 0 0 5 0 】

図 1 1 ( C ) は、R A M 1 2 0 において O S 管理領域に割り当てられる範囲が 0 ~ 2 . 0 G B の範囲まで縮小されるとともに、O S 管理外領域に R A M ディスク 1 2 2 と S S D 用キャッシュ領域 1 2 4 とが設定された状態のアドレス空間を示している。R A M ディスクドライバー 1 1 2 は、S S D 用キャッシュ領域 1 2 4 の容量が既知である場合には、R A M ディスク 1 2 2 の容量と S S D 用キャッシュ領域 1 2 4 の容量との和に応じて、R A M 1 2 0 における O S 管理領域のメモリ範囲を縮小させるものとしても良い。あるいは、R A M ディスクドライバー 1 1 2 は、S S D 用キャッシュ領域 1 2 4 の少なくとも一部が R A M 1 2 0 の O S 管理外領域に設定できるように、R A M 1 2 0 における O S 管理領域のメモリ範囲を余分に縮小するものとしても良い。

## 【 0 0 5 1 】

このように、ステップ S 1 2 0 ( 図 1 0 ) において、S S D 用キャッシュ領域 1 2 4 が R A M 1 2 0 上に設定されると、S S D ドライバー 1 1 4 は、S S D 1 3 0 A へのデータの書込要求の有無の監視を開始する ( ステップ S 1 3 0 ) 。 S S D ドライバー 1 1 4 は、データの書込要求を検出すると、S S D 用キャッシュ領域 1 2 4 に当該データを格納する ( ステップ S 1 4 0 ) 。ここで、S S D 1 3 0 A に直接データを書き込む速度より、R A M 1 2 0 の S S D 用キャッシュ領域 1 2 4 にデータを書き込む速度の方が速いため、ユーザーは S S D 1 3 0 A へのデータの書き込み速度が向上しているように感じる。

## 【 0 0 5 2 】

ステップ S 1 5 0 では、S S D ドライバー 1 1 4 は、規定のデータ量が S S D 用キャッシュ領域 1 2 4 に格納されているか否かを判定する。S S D 用キャッシュ領域 1 2 4 に規定量のデータが格納されている場合には、S S D ドライバー 1 1 4 は、S S D 用キャッシュ領域 1 2 4 に格納されているデータを S S D 1 3 0 A に転送する ( ステップ S 1 6 0 ) 。 S S D 用キャッシュ領域 1 2 4 に規定量のデータが格納されていなかった場合には、S S D ドライバー 1 1 4 は、S S D 1 3 0 A へのデータの書込要求の有無の監視を再開する ( ステップ S 1 3 0 ) 。また、ステップ S 1 6 0 において、データの転送が完了した場合にも、S S D ドライバー 1 1 4 は、S S D 1 3 0 A へのデータの書込要求の有無の監視を再開する。

## 【 0 0 5 3 】

ステップ S 1 3 0 ~ S 1 6 0 の処理によって、S S D 用キャッシュ領域 1 2 4 は、S S D 1 3 0 A のディスクキャッシュとして機能する。即ち、S S D 用キャッシュ領域 1 2 4 によって、ユーザーが体感する S S D 1 3 0 A へのデータ転送速度が向上する。また、S S D 1 3 0 A へのアクセス回数が低減され、S S D 1 3 0 A に対するアクセス効率が向上する。

## 【 0 0 5 4 】

このように、第 2 実施例のコンピューター 1 0 0 A であれば、R A M 1 2 0 の O S 管理外領域に、R A M ディスク 1 2 2 と S S D 用キャッシュ領域 1 2 4 とを、互いの領域が重複しない連続領域として設けることが可能である。従って、R A M ディスク 1 2 2 と S S D 用キャッシュ領域 1 2 4 とに対するアクセス速度が向上する。

## 【 0 0 5 5 】

C . 変形例 :

なお、この発明は上記の実施例や実施形態に限られるものではなく、その要旨を逸脱し

10

20

30

40

50

ない範囲において種々の態様において実施することが可能であり、例えば次のような変形も可能である。

【 0 0 5 6 】

C1．変形例 1：

上記実施例において、ハードウェアによって実現されていた構成の一部をソフトウェアに置き換えるようにしてもよく、逆に、ソフトウェアによって実現されていた構成の一部をハードウェアに置き換えるようにしてもよい。また、RAMディスクドライバ 112 の機能の一部を他のプログラムやハードウェアが実行するようにすることもできる。

【 0 0 5 7 】

C2．変形例 2：

上記実施例において、コンピューター 100, 100A には、32ビットOSが導入されていた。しかし、OSとしては、64ビットOSが導入されるものとしても良い。ここで、64ビットOSでは、一般に、RAMの物理メモリのうち、8GB～128GBまでの範囲がOS管理領域として設定されるが、この場合には、物理メモリのうち、その設定範囲の外側の領域がOS管理外領域となる。

【 0 0 5 8 】

C3．変形例 3：

上記実施例では、RAMディスクドライバ 112 は、ユーザーが指定したRAMディスク 122 の容量が、RAM 120 におけるOS管理外領域の容量より大きいときに、RAM 120 におけるOS管理領域のメモリ範囲を縮小させていた。しかし、RAMディスクドライバ 112 は、例えば、ユーザーが指定したRAMディスク 122 の容量が、RAM 120 におけるOS管理外領域の容量より小さいときに、RAM 120 上のOS管理外領域における未使用領域が低減されるように、OS管理領域の範囲を増大させるものとしても良い。

【 0 0 5 9 】

C4．変形例 4：

上記実施例では、RAMディスクドライバ 112 は、OS管理外領域の末尾からの連続領域をRAMディスク 122 の記憶領域として設定していた。しかし、RAMディスクドライバ 112 は、OS管理外領域の中の任意の位置の連続領域を、RAMディスク 122 の記憶領域として設定するものとしても良い。

【 0 0 6 0 】

C5．変形例 5：

上記実施例では、RAMディスクドライバ 112 は、boot.ini ファイルに /MAXMEM = nn スイッチを設けることにより、RAM 120 におけるOS管理領域の範囲を変更していた。しかし、RAMディスクドライバ 112 は、他の手段によって、OS管理領域の範囲を変更するものとしても良い。例えば、OSがWindows XP である場合には、boot.ini ファイルに /BURNMEMORY = nn スイッチを設けて、OSが使用できないメモリ容量を指定するものとしても良い。また、OSがWindows Vista や Windows 7 である場合には、RAMディスクドライバ 112 は、ブート構成データを編集して、RAM 120 におけるOS管理領域の範囲を変更するものとしても良い。なお、「WINDOWS VISTA」はマイクロソフトコーポレーションの登録商標である。

【 0 0 6 1 】

C6．変形例 6：

上記実施例では、RAMディスクドライバ 112 は、操作画面 10 を介して、ユーザーからのRAMディスク 122 の容量の指定を受け付けていた。しかし、RAMディスクドライバ 112 は、操作画面 10 を介して、ユーザーからRAM 120 におけるOS管理領域の範囲の指定を受け付けることができるものとしても良い。

【 0 0 6 2 】

C7．変形例 7：

上記第2実施例では、RAMディスクドライバー112が、RAMディスク122として使用しないOS管理外領域の空き領域に関する情報をレジストリに格納し、SSDドライバー114が、レジストリから当該情報を取得していた。しかし、RAMディスクドライバー112は、当該情報を他の手段によって、SSDドライバー114に取得させるものとしても良い。例えば、RAMディスクドライバー112とSSDドライバー114との間で予め規定されたSSD130Aにおけるレジストリ以外の格納場所に、当該情報を格納するものとしても良い。

【0063】

C8. 変形例8:

上記第2実施例では、RAMディスクドライバー112がRAM120のOS管理外領域に対して優先的にRAMディスク122を設定していた。しかし、SSDドライバー114が、第2実施例で説明したRAMディスクドライバー112の処理を実行することにより、RAM120のOS管理外領域にRAMディスク112より優先させてSSD用キャッシュ領域124が設定されるものとしても良い。

【0064】

C9. 変形例9:

上記第2実施例では、コンピューター100Aは、外部記憶装置としてSSD130Aを備え、そのデバイスドライバーとして、SSDドライバー114を有していた。しかし、コンピューター100Aは、外部記憶装置としてSSD以外の他の種類の記憶装置を備え、当該記憶装置のためのデバイスドライバーを備えるものとしても良い。この場合には、当該デバイスドライバーが、RAM120のOS管理外領域に当該記憶装置のためのキャッシュ領域を設定するものとしても良い。

【0065】

C10. 変形例10:

上記第2実施例では、SSDドライバー114が、RAMディスクドライバー112によってレジストリに書き込まれた情報を読み込み、その情報に基づいて、RAM120のOS管理外領域を使用していた。しかし、SSDドライバー114に換えて、他のソフトウェアが、RAMディスクドライバー112が書き込んだレジストリの情報を読み込み、その情報に基づいて、RAM120のOS管理外領域を使用するものとしても良い。

【符号の説明】

【0066】

- 10 ... 操作画像
- 11, 12 ... 第1と第2のレベルバー
- 11c, 12c ... 第1と第2のチェックボックス
- 13 ... スライダー
- 14 ... 決定ボタン
- 15 ... 終了ボタン
- 100, 100A ... コンピューター
- 101 ... 内部バス
- 110 ... CPU
- 112 ... RAMディスクドライバー
- 114 ... SSDドライバー
- 120 ... RAM
- 122 ... RAMディスク
- 124 ... SSD用キャッシュ領域
- 130 ... 外部記憶装置(HDD)
- 130A ... 外部記憶装置(SSD)
- 140 ... 表示部
- 150 ... 操作部

10

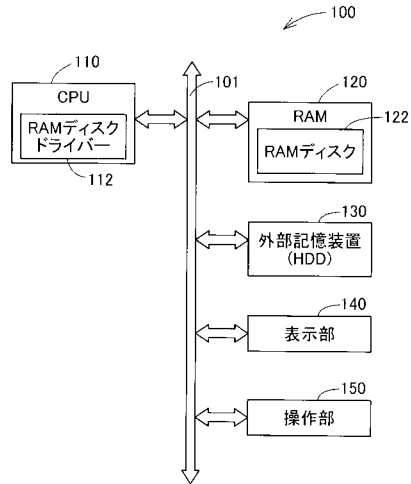
20

30

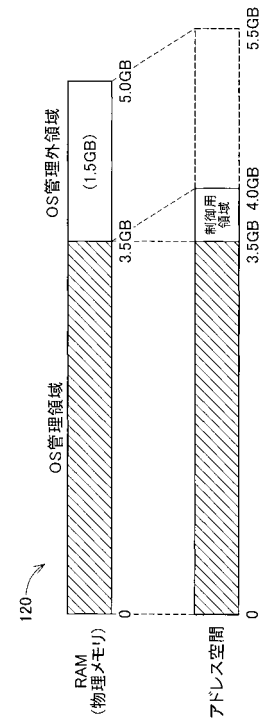
40

【図 1】

第1実施例

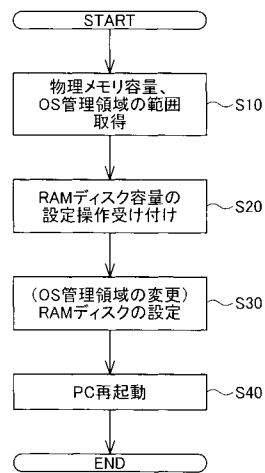


【図 2】



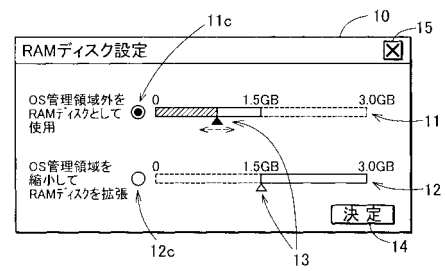
【図 3】

第1実施例

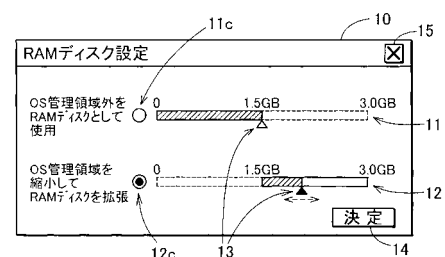


【図 4】

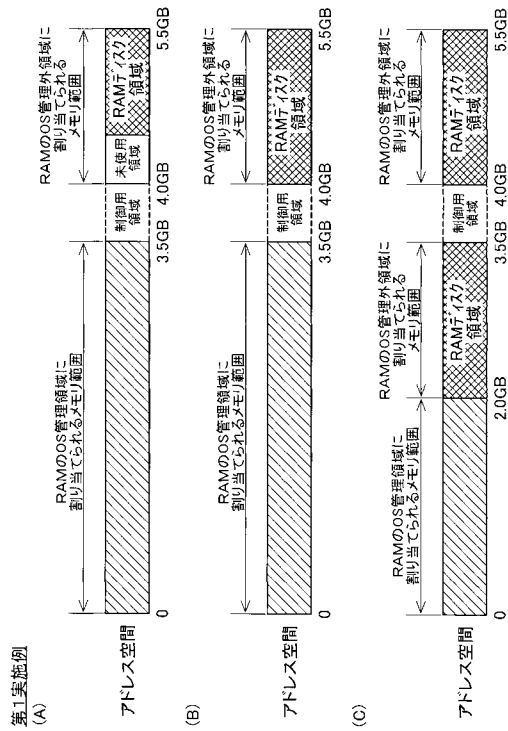
(A)



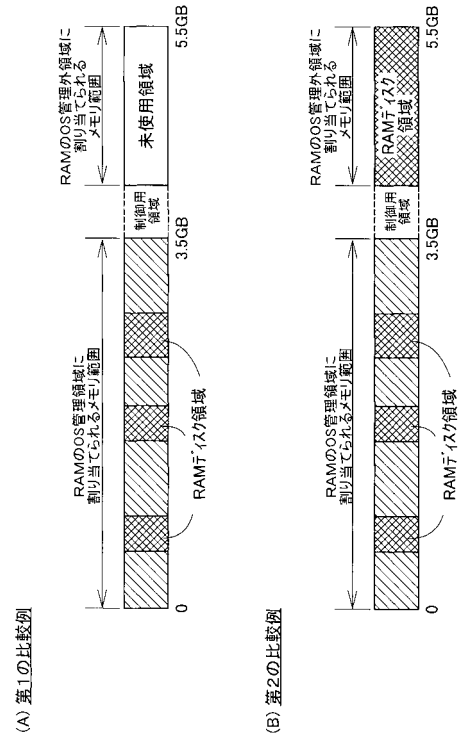
(B)



【 図 5 】

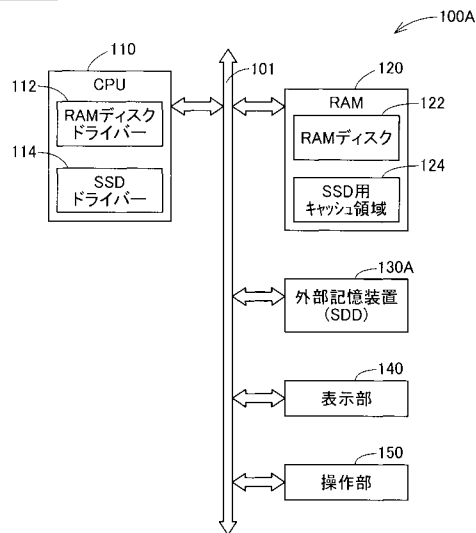


【 図 6 】



【 図 7 】

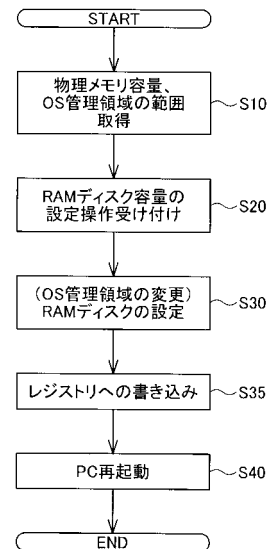
## 第2实施例



【 図 8 】

## 第2実施例

### RAMディスクドライバ



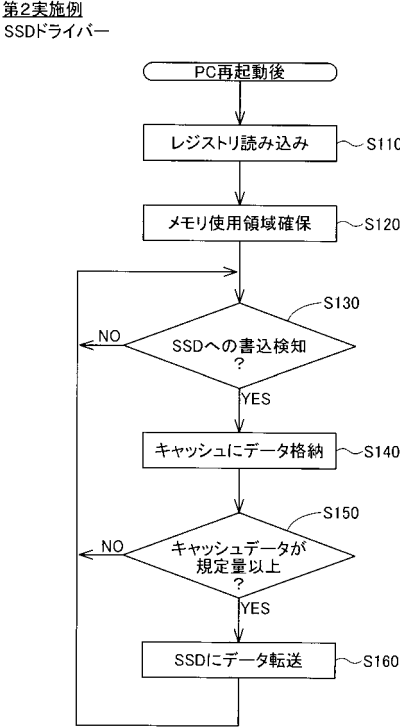
【図 9】

第2実施例

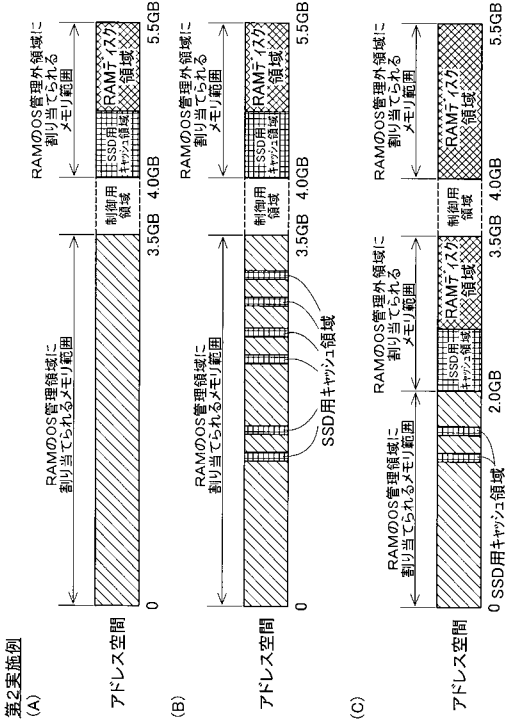
レジストリ設定例

名 称	型	内 容
EnableOver4G	REG_DWORD	0 : OS管理外領域は使用不可 1 : OS管理外領域は使用可能
StartAddr	REG_DWORD	SSD用キャッシュ領域として 使用できる領域の先頭物理アドレス
CacheSize	REG_DWORD	SSD用キャッシュ領域として 使用できる容量

【図 10】



【図 11】





---

フロントページの続き

- (56)参考文献 平澤寿康, 【特別レポート】32bit Windowsの管理外領域をRAM Diskに使う, P C W a t c h [online], 2008年 5月12日, p. 1 - 4, インターネット<URL: <http://pc.watch.impress.co.jp/docs/2008/0512/ramdisk.htm>>  
いま持っているパソコンをラムディスクで速くしよう!, ビギナーのための周辺機器情報サイト おしえて! 周辺機器 [online], 2009年12月, p. 1 - 5, インターネット<URL: <http://buffalo.jp/products/oshiete/special/ramdisk/>>

- (58)調査した分野(Int.Cl., DB名)

G 0 6 F 1 2 / 0 0 - 1 2 / 0 6  
G 0 6 F 1 3 / 1 6 - 1 3 / 1 8