

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4725035号  
(P4725035)

(45) 発行日 平成23年7月13日(2011.7.13)

(24) 登録日 平成23年4月22日(2011.4.22)

(51) Int. Cl. F 1  
**G 0 6 F 12/00 (2006.01)** G O 6 F 12/00 5 3 1 M  
**G 0 6 F 3/06 (2006.01)** G O 6 F 3/06 3 0 4 F

請求項の数 7 (全 27 頁)

(21) 出願番号	特願2004-162744 (P2004-162744)	(73) 特許権者	000002897
(22) 出願日	平成16年6月1日(2004.6.1)		大日本印刷株式会社
(65) 公開番号	特開2005-346219 (P2005-346219A)		東京都新宿区市谷加賀町一丁目1番1号
(43) 公開日	平成17年12月15日(2005.12.15)	(74) 代理人	100091476
審査請求日	平成19年3月5日(2007.3.5)		弁理士 志村 浩
		(72) 発明者	近田 恭之
			東京都新宿区市谷加賀町一丁目1番1号
			大日本印刷株式会社内
		(72) 発明者	庭田 尚蔵
			東京都新宿区市谷加賀町一丁目1番1号
			大日本印刷株式会社内
		(72) 発明者	矢野 義博
			東京都新宿区市谷加賀町一丁目1番1号
			大日本印刷株式会社内

最終頁に続く

(54) 【発明の名称】 データバックアップ装置

(57) 【特許請求の範囲】

【請求項1】

コンピュータに組み込まれて用いられるデータバックアップ装置であって、  
 前記コンピュータのメモリ上に展開されているファイルを第1の格納場所に所定のファイル名で上書き保存する保存処理が行われたことを検出する保存処理検出手段と、  
 前記第1の格納場所に保存されている個々のファイルごとに、保存処理が行われた時点からの経過時間をそれぞれ計時値としてカウントする計時手段と、  
 特定のファイルについて前記第1の格納場所への保存処理が行われるたびに、当該特定のファイルについての計時値を0にリセットして、前記計時手段に0からカウントを開始させる計時リセット手段と、

特定のファイルについての計時値が所定値Tに達したら、当該特定のファイルを前記第1の格納場所から第2の格納場所に転送し、前記第2の格納場所に既に保存されている既存のファイルとは別個のファイルとして保存してバックアップを行うとともに、当該特定のファイルについての計時値のカウントを停止させるバックアップ実行手段と、

を備え、同一のファイル名で特定されるファイルの異なる更新時点の内容を示す複数のファイルが前記第2の格納場所に保存されるようにすることを特徴とするデータバックアップ装置。

【請求項2】

請求項1に記載のデータバックアップ装置において、  
 バックアップ実行手段が、コンピュータのシステム終了時に、計時手段によるカウント

が継続して行われている計時継続ファイルを認識し、この計時継続ファイルを第1の格納場所から第2の格納場所にバックアップする機能を有することを特徴とするデータバックアップ装置。

【請求項3】

コンピュータに組み込まれて用いられるデータバックアップ装置であって、

前記コンピュータのメモリ上に展開されているファイルを第1の格納場所に所定のファイル名で上書き保存する保存処理が行われたことを検出する保存処理検出手段と、

前記第1の格納場所に保存されている個々のファイルごとに、所定時点からの経過時間をそれぞれ計時値としてカウントする計時手段と、

前記コンピュータのシステム起動後、所定のファイルについて前記第1の格納場所への保存処理が初めて行われたときに、当該所定のファイルについて、前記計時手段に0からカウントを開始させる計時開始手段と、

特定のファイルについての計時値が所定値Tに達したら、当該計時値を所定の設定値にリセットして、前記計時手段にカウントを継続させるとともに、その時点で前記第1の格納場所に保存されている当該特定のファイルに対するバックアップが実行済か否かを認識し、実行済でなかった場合には、当該特定のファイルを第2の格納場所に転送し、前記第2の格納場所に既に保存されている既存のファイルとは別個のファイルとして保存するバックアップを行うバックアップ実行手段と、

を備え、同一のファイル名で特定されるファイルの異なる更新時点の内容を示す複数のファイルが前記第2の格納場所に保存されるようにし、

前記バックアップ実行手段が、前記特定のファイルについての計時値が所定値Tに達したときに、前記特定のファイルの最終保存時刻から現在時刻までの経過時間を認識し、前記経過時間が所定値Tであった場合には、計時値を0にリセットし、前記経過時間が所定値Tでなかった場合には、計時値を前記経過時間に相当する値にリセットすることを特徴とするデータバックアップ装置。

【請求項4】

請求項3に記載のデータバックアップ装置において、

バックアップ実行手段が、コンピュータのシステム終了時に、第1の格納場所に保存されているファイルのうち、バックアップが実行済でない未バックアップファイルを認識し、この未バックアップファイルを第1の格納場所から第2の格納場所に転送し、前記第2の格納場所に既に保存されている既存のファイルとは別個のファイルとして保存するバックアップを行う機能を有することを特徴とするデータバックアップ装置。

【請求項5】

請求項1～4のいずれかに記載のデータバックアップ装置において、

バックアップ対象データを、ボリューム単位、フォルダ単位、もしくはファイル単位で、予め設定できるようにしておき、計時値のカウントをバックアップ対象データについてのみ行うようにしたことを特徴とするデータバックアップ装置。

【請求項6】

請求項1～5のいずれかに記載のデータバックアップ装置において、

バックアップ実行手段が、特定のファイルについてバックアップを実行する際に、前回のバックアップ時の内容に対する差分データのみを第2の格納場所に保存する処理を行うことを特徴とするデータバックアップ装置。

【請求項7】

請求項1～6のいずれかに記載のデータバックアップ装置としてコンピュータを機能させるためのプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、データバックアップ装置に関し、特に、内容が頻繁に更新されるファイルについて、自動的にデータのバックアップ処理を実行するデータバックアップ装置に関する

10

20

30

40

50

## 【背景技術】

## 【0002】

コンピュータを利用するにあたって、データのバックアップ作業は非常に重要な事項である。特に、業務用コンピュータを利用する上では、定期的にデータをバックアップする作業が不可欠になる。バックアップ用データを格納するデータ記録媒体としては、ハードディスク、光ディスク、光磁気ディスク、磁気テープなど、様々なものが利用されているが、バックアップの基本原理は、同一の情報を複数の記録媒体に格納しておくことにある。したがって、バックアップ作業の本質は、バックアップ対象となるデータを、1つの格納場所から別な格納場所に複製する複製処理に他ならない。

10

## 【0003】

通常、コンピュータのOSプログラムには、所望のファイルを複製するためのコピーコマンドが含まれており、オペレータは、このコピーコマンドを実行させることにより、随時、任意のファイルに対するバックアップ作業を行うことができる。また、アプリケーションプログラムによっては、ボタン1つでバックアップ作業を実行する機能を備えるものもある。

## 【0004】

しかしながら、重要なバックアップ作業を、逐次、オペレータの指示に委ねることは、実用上、好ましくない。オペレータにとって、バックアップ対象となるデータを確認し、当該データについてのバックアップ操作を行う作業は、きわめて煩雑な作業であり、失念する可能性も少なくない。このため、バックアップ作業を自動的に実行するデータバックアップ装置が普及している。一般的な自動データバックアップ装置は、コンピュータに、専用のバックアップ用アプリケーションプログラムを組み込むことにより構築され、予め設定した所定のスケジュールに基づいて、予め設定した所定のデータに対するバックアップ処理を自動的に実行する機能を有している。たとえば、下記の特許文献1, 2には、スケジュールリング機能を備えた自動データバックアップ装置が開示されている。

20

【特許文献1】特開2001-249851号公報

【特許文献2】特開2003-263354号公報

## 【発明の開示】

## 【発明が解決しようとする課題】

30

## 【0005】

オペレータが、アプリケーションプログラムを利用して、データファイルに対して何らかの作業を実行する場合、作業の途中段階でファイルの保存操作を何度か行うのが一般的である。これは、ある程度作業が進行した段階で、その時点でのファイルの内容をメモリからハードディスク装置などの記憶装置に保存しておく方が安全であるためである。したがって、ハードディスク装置などに保存されているファイルの内容は、オペレータが保存操作を行うごとに更新されることになる。アプリケーションプログラムによっては、オペレータの保存操作を待たずに、所定の条件が満たされるたびに、自動的に保存処理を実行することもある。

## 【0006】

40

このように、内容が頻繁に更新されるファイルについてデータのバックアップを行う場合、バックアップ処理のタイミングが問題になる。安全性を第一に考えれば、オペレータの保存操作により、ファイルの内容が更新されるたびに、毎回、バックアップを行うのが好ましい。しかしながら、頻繁に更新されるファイルを、それぞれ更新のたびにバックアップすることになると、膨大なバックアップ容量を確保する必要が生じる。また、バックアップ処理中は、CPUに負担がかかるため、頻繁にバックアップ処理が実行されると、コンピュータの応答性が悪くなるという問題も生じる。更に、万一、バックアップしてあったデータを復元する必要が生じた場合、更新のたびにバックアップを行っているため、膨大な数の復元ポイントが存在することになるため、適切な復元ポイントを選択することが非常に繁雑になる。

50

## 【 0 0 0 7 】

このような弊害に対処するために、一定の時間ごとに自動的にバックアップする、というような手法を採ることも可能であるが、ファイルの保存操作とは無関係にバックアップが行われることになるため、効率的な手法とは言えない。

## 【 0 0 0 8 】

そこで本発明は、内容が頻繁に更新されるファイルについて、効率的なバックアップ処理を実行することが可能なデータバックアップ装置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 9 】

(1) 本発明の第1の態様は、コンピュータに組み込まれて用いられるデータバックアップ装置において、

コンピュータのメモリ上に展開されているファイルを第1の格納場所に所定のファイル名で上書き保存する保存処理が行われたことを検出する保存処理検出手段と、

第1の格納場所に保存されている個々のファイルごとに、保存処理が行われた時点からの経過時間をそれぞれ計時値としてカウントする計時手段と、

特定のファイルについて第1の格納場所への保存処理が行われるたびに、当該特定のファイルについての計時値を0にリセットして、計時手段に0からカウントを開始させる計時リセット手段と、

特定のファイルについての計時値が所定値Tに達したら、当該特定のファイルを第1の格納場所から第2の格納場所に転送し、第2の格納場所に既に保存されている既存のファイルとは別個のファイルとして保存してバックアップを行うとともに、当該特定のファイルについての計時値のカウントを停止させるバックアップ実行手段と、

によってデータバックアップ装置を構成し、同一のファイル名で特定されるファイルの異なる更新時点の内容を示す複数のファイルが前記第2の格納場所に保存されるようにしたものである。

## 【 0 0 1 0 】

(2) 本発明の第2の態様は、上述の第1の態様に係るデータバックアップ装置において、

バックアップ実行手段が、コンピュータのシステム終了時に、計時手段によるカウントが継続して行われている計時継続ファイルを認識し、この計時継続ファイルを第1の格納場所から第2の格納場所にバックアップする機能を有するようにしたものである。

## 【 0 0 1 1 】

(3) 本発明の第3の態様は、コンピュータに組み込まれて用いられるデータバックアップ装置において、

コンピュータのメモリ上に展開されているファイルを第1の格納場所に所定のファイル名で上書き保存する保存処理が行われたことを検出する保存処理検出手段と、

第1の格納場所に保存されている個々のファイルごとに、所定時点からの経過時間をそれぞれ計時値としてカウントする計時手段と、

コンピュータのシステム起動後、所定のファイルについて第1の格納場所への保存処理が初めて行われたときに、当該所定のファイルについて、計時手段に0からカウントを開始させる計時開始手段と、

特定のファイルについての計時値が所定値Tに達したら、当該計時値を所定の設定値にリセットして、計時手段にカウントを継続させるとともに、その時点で第1の格納場所に保存されている当該特定のファイルに対するバックアップが実行済か否かを認識し、実行済でなかった場合には、当該特定のファイルを第2の格納場所に転送し、第2の格納場所に既に保存されている既存のファイルとは別個のファイルとして保存するバックアップを行うバックアップ実行手段と、

によってデータバックアップ装置を構成し、同一のファイル名で特定されるファイルの異なる更新時点の内容を示す複数のファイルが前記第2の格納場所に保存されるようにし

バックアップ実行手段が、特定のファイルについての計時値が所定値Tに達したときに、当該特定のファイルの最終保存時刻から現在時刻までの経過時間を認識し、経過時間が所定値Tであった場合には、計時値を0にリセットし、経過時間が所定値Tでなかった場合には、計時値を当該経過時間に相当する値にリセットするようにしたものである。

【0012】

(4) 本発明の第4の態様は、上述の第3の態様に係るデータバックアップ装置において、

バックアップ実行手段が、コンピュータのシステム終了時に、第1の格納場所に保存されているファイルのうち、バックアップが実行済でない未バックアップファイルを認識し、この未バックアップファイルを第1の格納場所から第2の格納場所に転送し、第2の格納場所に既に保存されている既存のファイルとは別個のファイルとして保存するバックアップを行う機能を有するようにしたものである。

10

【0014】

(5) 本発明の第5の態様は、上述の第1～第4の態様に係るデータバックアップ装置において、

バックアップ対象データを、ボリューム単位、フォルダ単位、もしくはファイル単位で、予め設定できるようにしておき、計時値のカウントをバックアップ対象データについてのみ行うようにしたものである。

【0015】

(6) 本発明の第6の態様は、上述の第1～第5の態様に係るデータバックアップ装置において、

20

バックアップ実行手段が、特定のファイルについてバックアップを実行する際に、前回のバックアップ時の内容に対する差分データのみを第2の格納場所に保存する処理を行うようにしたものである。

【0016】

(7) 本発明の第7の態様は、上述の第1～第6の態様に係るデータバックアップ装置としてコンピュータを機能させるためのプログラムを用意し、当該プログラムをコンピュータ読取り可能な記録媒体に記録して配付できるようにしたものである。

【発明の効果】

【0017】

30

本発明に係るデータバックアップ装置によれば、バックアップ対象となる個々のファイルごとに、保存処理時からの経過時間をカウントし、このカウント値に応じた適切なタイミングでバックアップ処理を実行するようにしたため、内容が頻繁に更新されるファイルについても、効率的なバックアップ処理を実行することが可能になる。

【発明を実施するための最良の形態】

【0018】

以下、本発明を図示する実施形態に基づいて説明する。

【0019】

<<< §1. データバックアップ装置の基本構成 >>>

図1は、本発明の第1の実施形態に係るデータバックアップ装置を組み込んだコンピュータシステムの基本構成を示すブロック図である。図に示すメモリ10、第1の格納場所20、処理実行部30は、一般的なコンピュータが標準的に備えている基本構成要素である。すなわち、メモリ10は、通常、揮発性メモリであるRAMによって構成され、実行中のプログラムやデータを展開する領域として利用される。第1の格納場所20は、プログラムやデータを保存する主たる記憶装置によって構成され、図示の例の場合、コンピュータの内蔵ハードディスクからなる。また、処理実行部30は、このコンピュータの中核をなすCPUや入出力インターフェイス、基本プログラムを格納したROMなどを含む構成要素であり、オペレータの指示やプログラムに基づいて、様々な処理作業を行う。なお、一般的なコンピュータには、マウス・キーボードなどの入力機器や、ディスプレイ・プリンタなどの出力機器が接続されるのが普通であるが、ここでは本発明の動作に直接関与

40

50

しない構成要素については図示を省略している。

【0020】

一方、第2の格納場所40は、第1の格納場所20に保存されているデータをバックアップするための記憶装置であり、図示の例の場合、バックアップサーバーによって構成されている。処理実行部30は、ネットワーク通信線を介して、第2の格納場所40を構成するバックアップサーバーとデータのやりとりを行うことができ、第1の格納場所20内に保存されている特定のファイルの複製を、第2の格納場所40へ保存することにより、当該ファイルに対するバックアップを行うことができる。

【0021】

データバックアップ装置50は、本発明の対象となる装置そのものであり、図示のとおり、保存処理検出手段51、計時リセット手段52、計時手段53、バックアップ実行手段54によって構成されている。もっとも、これら各手段は、後述するように、それぞれコンピュータ上で特定の処理を行う機能要素であり、実際には、コンピュータのハードウェアとソフトウェアの組み合わせによって実現されるものである。別言すれば、本発明に係るデータバックアップ装置50（第2の実施形態として述べるデータバックアップ装置60も同様）は、汎用コンピュータに専用のバックアップ用プログラムをインストールすることにより、当該汎用コンピュータの内部に具現化される装置である。もちろん、このようなバックアップ用プログラムは、特許法上の物に該当する経済価値を有する産物であり、ネットワークを介して配付することも可能であるし、CD-ROMなどのコンピュータ読み取り可能な記録媒体に記録した状態で配付することも可能である。

【0022】

なお、図1に示す実施形態では、説明の便宜上、一点鎖線で囲った部分を、1台のコンピュータの筐体内に組み込まれた構成要素として捉えており、第2の格納場所40のみが、当該コンピュータの外部の構成要素となっているが、このような構成を採ることは、本発明を実施するにあたって、本質的な問題ではない。たとえば、第1の格納場所20を、USBケーブルなどでコンピュータの筐体外に接続したハードディスク装置によって構成してもよいし、逆に、第2の格納場所40を、コンピュータの筐体内に組み込んだ第2のハードディスク装置によって構成してもよい。もちろん、第1の格納場所20や第2の格納場所40は、必ずしもハードディスク装置で構成する必要はなく、光ディスク装置、光磁気ディスク装置、磁気テープ記録装置などによって構成してもかまわない。

【0023】

第1の格納場所20には、通常、多数のファイルが保存されている。図示の例では、便宜上、3つのファイルF a , F b , F cだけが、第1の格納場所20に保存された状態が示されているが、実際には、より多数のファイルが保存されている。これらのファイルは、データファイルであったり、プログラムファイルであったりする。処理実行部30は、第1の格納場所20（あるいは、別な格納場所でもかまわない）から、所定のプログラムファイルやデータファイルを読み込み、これをメモリ10上に展開した上で、様々な処理を実行する。図では、メモリ10上に、2つのファイルF a , F cが展開されている状態が示されている。

【0024】

以下、説明の便宜上、ファイルF a , F b , F cが、いずれも所定のワープロ用アプリケーションプログラムで作成されたデータファイルであるものとしよう。図示の例では、このワープロ用の2つのファイルF a , F cがメモリ10上に展開されているが、これは、当該ワープロ用アプリケーションプログラムを用いて、2つのファイルF a , F cを開く指示操作を行った状態に対応する。処理実行部30は、オペレータからの操作入力に基づき、ファイルF a , F cの内容をメモリ10上で編集する処理を行う。このような編集処理を行えば、メモリ10上に展開されている各ファイルF a , F cのデータは更新されることになるが、その時点では、第1の格納場所20に保存されている各ファイルF a , F cは、更新前のままである。

【0025】

そこで、オペレータは、ひととおりの編集作業が完了した時点で、当該ワープロ用アプリケーションプログラムに対して、ファイルF a , F c の保存指示を与える。このとき、元と同じファイル名で保存する指示を与えれば、第1の格納場所20に保存されていた更新前の古いファイルF a , F c は、メモリ10内の更新後の新しいファイルF a , F c に置き換えられる(いわゆる、上書き保存)。

【0026】

メモリ10を構成するRAMは、揮発性メモリであるので、コンピュータの電源を落とすと、展開されていたファイルF a , F c の情報は失われてしまう。したがって、各ファイルF a , F c に対する編集作業が完了したら、これらを第1の格納場所20に保存した後に作業を完了することは、オペレータにとって当然の手順である。ただ、実用上は、各ファイルF a , F c に対する編集作業が完了するまでの途中段階においても、オペレータは、保存作業を頻繁に行うことが多い。これは、作業途中において、万一、アプリケーションプログラムやOSプログラムが動作不良(いわゆる、プログラムの暴走)を生じた場合のリスクを軽減するためである。すなわち、作業途中において、プログラムが動作不良を生じると、通常、メモリ10上に展開されているファイルを救済することは困難になるので、オペレータは、編集作業の途中段階であっても、随時、その時点までの更新内容を含むファイルを保存する操作を行い、万一の事態に備えるのである。アプリケーションプログラムによっては、オペレータの保存操作を待たずに、所定の条件が満たされるたびに、自動的に保存処理を実行することもある。

【0027】

こうして第1の格納場所20に保存された情報は、装置が正常な状態を維持している限り、失われることはないが、100%の安全性が保証されているわけではない。実際、ハードディスク装置には機器固有の寿命が存在し、保存した情報が突然読み出せなくなる事態が生じる危険性がある。このような危険を回避するために、データのバックアップを行うことになる。すなわち、第1の格納場所20に保存されている各ファイルの複製を、第2の格納場所40にも作成しておくことにより、万一、一方のファイルが読出し不能になっても、もう一方のファイルに基づいて復元できるような安全策が採られることになる。

【0028】

このように、本発明に係るデータバックアップ装置によるバックアップ処理とは、第1の格納場所20に保存されているファイルのデータを、第2の格納場所40へ転送し、その複製を作成する処理に他ならない。図示の例では、第2の格納場所40内に、ファイルF a , F b , F c の複製が保存されている状態が示されている。

【0029】

ところが、内容が頻繁に更新されるファイルについてバックアップを行う場合、バックアップ処理のタイミングが問題になる。安全性を第一に考えれば、オペレータの保存操作により、ファイルの内容が更新されるたびに、毎回、バックアップを行うのが好ましいが、その場合、前述したとおり様々な弊害が生じることになる。

【0030】

たとえば、図1に示す例において、ワープロ用データファイルF a に対する編集作業を行っているオペレータが、3分おきに保存操作を行ったとすると、1時間の間に、20通りもの更新ファイルが作成されることになるので、これらをすべて第2の格納場所40にバックアップすると、膨大なバックアップ容量を確保する必要が生じる。また、この1時間の作業だけで、20通りの復元ポイントが生まれることになるので、万一の際に復元作業を行うにしても、適切な復元ポイントを選択することが非常に繁雑になる。更に、処理実行部30がバックアップ処理を実行している最中は、他のタスクの処理効率が低下するため、上述の例の場合、3分ごとにコンピュータの応答性が低下する事態が生じることになる。

【0031】

本発明は、このような点に着目し、内容が頻繁に更新されるファイルについて、効率的なバックアップ処理を実行するための新たな手法を提示するものである。

## 【 0 0 3 2 】

< < < § 2 . 本発明の第 1 の実施形態 > > >

図 1 に示すデータバックアップ装置 5 0 は、本発明の第 1 の実施形態に係る装置であるが、その基本原理を図 2 のタイムチャートを参照しながら説明する。このタイムチャートの中心線は、図の上から下に向かっての時間の流れを示す時間軸になっており、その左側には、オペレータの操作タイミングが示され、右側には、データバックアップ装置 5 0 によるバックアップ処理のタイミングが示されている。

## 【 0 0 3 3 】

具体的には、オペレータは、まず、時刻  $t_1$  において、所定のアプリケーションプログラムを用いて、ファイルを開く操作を行い、以後、当該ファイルに対する編集作業を行っている。しかも、このオペレータは、編集作業の途中段階の時刻  $t_2$  ,  $t_3$  ,  $t_4$  ,  $t_6$  ,  $t_7$  において、編集途中のファイルを保存する操作を行い、最終的に、時刻  $t_8$  において、当該ファイルを閉じる操作を行っている。

10

## 【 0 0 3 4 】

なお、ここでは、便宜上、各更新段階におけるファイルを、それぞれ更新ファイル  $F(0)$  ~  $F(5)$ 、もしくは単にファイル  $F(0)$  ~  $F(5)$  と呼んで、かっこ内の番号 0 ~ 5 を用いて相互に区別することにするが、オペレータから見れば、各更新ファイル  $F(0)$  ~  $F(5)$  は、いずれもファイル名「F」で特定される同一のファイルであり、編集作業の途中段階の変遷を示すものに過ぎない。すなわち、オペレータは、時刻  $t_1$  において、第 1 の格納場所 2 0 に保存されているファイル名「F」なるファイルを開く操作（メモリ 1 0 上に展開する操作）を行い、当該ファイルに対して、メモリ 1 0 上で編集作業を行い、時刻  $t_2$  ,  $t_3$  ,  $t_4$  ,  $t_6$  ,  $t_7$  の各時点で、ファイル名「F」のまま上書き保存を行い、時刻  $t_8$  において、当該ファイルを閉じて作業を完了している。結局、第 1 の格納場所 2 0 には、ファイル名「F」をもった同一のファイルが常に格納されている状態ではあるが、ファイルの中身は、各時点ごとに上書きされて更新されることになる。

20

## 【 0 0 3 5 】

ここで、安全性を第 1 に考えれば、個々の更新ファイルを逐次バックアップするのが好ましい。図示の例の場合、ファイル  $F(0)$  については、通常であれば、前回既にバックアップが完了しているはずなので、時刻  $t_2$  で保存されたファイル  $F(1)$ 、時刻  $t_3$  で保存されたファイル  $F(2)$ 、時刻  $t_4$  で保存されたファイル  $F(3)$ 、時刻  $t_6$  で保存されたファイル  $F(4)$ 、時刻  $t_7$  で保存されたファイル  $F(5)$  のそれぞれについて、バックアップ処理を行い、第 2 の格納場所 4 0 に格納する処理を行えばよい。

30

## 【 0 0 3 6 】

しかしながら、本発明では、より効率的なバックアップ処理を実行するために、これらの更新ファイル  $F(1)$  ~  $F(5)$  のすべてについてバックアップを行うという方針を採る代わりに、選択されたいくつかについてのみバックアップを行うという方針を採る。具体的には、図 2 の例では、ファイル  $F(3)$  およびファイル  $F(5)$  のみがバックアップの対象となっており、ファイル  $F(1)$  ,  $F(2)$  ,  $F(4)$  についてのバックアップは省略されている。

40

## 【 0 0 3 7 】

5 つの更新ファイル  $F(1)$  ~  $F(5)$  のうち、2 つのファイル  $F(3)$  ,  $F(5)$  のみをバックアップの対象として選択した理由は、これらのファイルが保存された後、暫くの間、次の更新ファイルの保存が行われていないからである。たとえば、時刻  $t_2$  ,  $t_3$  ,  $t_4$  の時間間隔は比較的短時間であり、ファイル  $F(1)$  が保存された直後にファイル  $F(2)$  が保存され、その直後にファイル  $F(3)$  が保存されている。同様に、時刻  $t_6$  ,  $t_7$  の時間間隔も比較的短時間であり、ファイル  $F(4)$  が保存された直後にファイル  $F(5)$  が保存されている。これに対して、時刻  $t_4$  ,  $t_6$  の時間間隔は比較的長時間であり、更新ファイル  $F(3)$  が保存された後、次の更新ファイル  $F(4)$  が保存されるまでには、暫くの間が空いている。また、時刻  $t_7$  の後には、この例では保存作業が行われていないので、更新ファイル  $F(5)$  が保存された後、次の更新ファイルが保存されるま

50

では、無限大の時間が空いていることになる。

【 0 0 3 8 】

ここで述べる第 1 の実施形態の着眼点は、このように、「ある更新ファイル」が保存された後、暫くの間、次の更新ファイルの保存が行われなかった場合に、当該「ある更新ファイル」をバックアップの対象に選択する、という方針を採ることにより、効率的なバックアップを行う点にある。ここで、「暫くの間」をどの程度の時間設定にするかは、実用上の運用形態をみながら適宜設定すればよい。図 2 に示す実施形態は、この「暫くの間」を所定値 T に設定した例である。すなわち、時刻 t 2 ~ t 3 の間、時刻 t 3 ~ t 4 の間、時刻 t 6 ~ t 7 の間は、いずれも所定値 T よりも小さいため、更新ファイル F ( 1 ) , F ( 2 ) , F ( 4 ) はバックアップ対象として選択されていないが、時刻 t 4 ~ t 6 の間、時刻 t 7 ~ 次の保存時刻の間は、いずれも所定値 T よりも大きいため、ファイル F ( 3 ) , F ( 5 ) はバックアップの対象として選択されている。

10

【 0 0 3 9 】

このように、第 1 の更新ファイルの保存時刻と第 2 の更新ファイルの保存時刻との時間間隔が比較的小さい場合には、第 1 の更新ファイルをバックアップの対象とせず、両者の時間間隔が比較的大きい場合には、第 1 の更新ファイルをバックアップの対象とする、という取り扱いは、次のような理由により、道理にかなった取り扱いである。まず、第 1 に、比較的短い間隔で保存操作が繰り返し行われているときは、オペレータが入力操作を継続的に行っており、ひとまとまりの編集作業が進行中であると予想されるのに対し、保存操作が暫く行われていないときは、ひとまとまりの編集作業が一段落ついた状態であると

20

【 0 0 4 0 】

図 1 に示すデータバックアップ装置 5 0 は、上述した方針で、バックアップのタイミングを決める機能を果たすことになる。以下、このデータバックアップ装置 5 0 の各構成要素の機能を説明する。

【 0 0 4 1 】

まず、保存処理検出手段 5 1 は、コンピュータのメモリ 1 0 上に展開されているファイルを第 1 の格納場所 2 0 に所定のファイル名で保存する保存処理が行われたことを検出する構成要素である。通常、ファイルの保存処理は、コンピュータの OS プログラムの標準機能として備わっている処理であるので、保存処理検出手段 5 1 は、この OS プログラムの動作を監視するプログラムによって実現することができる。図 2 に示すタイムチャートの場合、保存処理検出手段 5 1 は、時刻 t 2 , t 3 , t 4 , t 6 , t 7 の各時点 ( 時間軸上の黒丸で示す時点 ) において保存処理が行われたことを検出することになる。

30

【 0 0 4 2 】

計時リセット手段 5 2 および計時手段 5 3 は、この保存処理検出手段 5 1 の検出結果に基づいて、計時処理を行うための構成要素である。すなわち、保存処理検出手段 5 1 が、ファイルの保存処理が行われたことを検出すると、計時リセット手段 5 2 は、その時点で、計時手段 5 3 の計時値を 0 にリセットし、0 からのカウントを開始させる。図 2 に示すタイムチャートの場合、時間軸上の黒丸で示す時点で、計時手段 5 3 に対するリセットが行われ、この時点をもととして、計時値のカウントを開始することになる。

40

【 0 0 4 3 】

バックアップ実行手段 5 4 は、計時手段 5 3 がカウントする計時値に応じて、バックアップ処理を実行する構成要素である。すなわち、計時値が所定値 T に達した時点で、バックアップ処理を実行するとともに、計時手段 5 3 による計時値のカウントを停止させる処理を行う。図 2 に示すタイムチャートの場合、時刻 t 2 で計時値 0 からのカウントが開始するが、所定値 T に達する前に時刻 t 3 で 0 にリセットされ、再度のカウントも、やはり

50

所定値 T に達する前に時刻 t 4 で 0 にリセットされてしまう。結局、計時値が最初に所定値 T に達するのは、時刻 t 5 ということになるので、バックアップ実行手段 5 4 によるバックアップ処理は、この時刻 t 5 の時点で実行される。このとき、第 1 の格納場所 2 0 内に保存されているファイルは、更新ファイル F ( 3 ) であるので、図示のとおり、時刻 t 5 において、ファイル F ( 3 ) についてのバックアップが実行されることになる。そして、この時刻 t 5 の時点で、計時手段 5 3 による計時値のカウン트가停止させられる。

【 0 0 4 4 】

図 2 の時間軸上の白丸は、バックアップ処理およびカウンタ停止処理が実行される時点を示している。計時手段 5 3 によるカウンタは、時刻 t 2 ~ t 5 まで実行されるが、時刻 t 5 において、一旦、中断することになる。ただ、次の時刻 t 6 において、更新ファイル F ( 4 ) の保存処理が行われると、計時リセット手段 5 2 からの指示により、計時手段 5 3 は、再び計時値 0 からのカウンタを開始する。この計時値は、所定値 T に達する前に時刻 t 7 で 0 にリセットされるが、やがて時刻 t 9 に達した時点で、所定値 T に達することになる。そこで、この時刻 t 9 において、バックアップ実行手段 5 4 によるバックアップ処理が実行され、その時点で第 1 の格納場所 2 0 内に保存されているファイル F ( 5 ) が第 2 の格納場所 4 0 へとバックアップされることになる。また、時刻 t 9 において、計時手段 5 3 による計時値のカウン트가停止させられる。

【 0 0 4 5 】

このように、ここで述べる実施形態の場合、計時手段 5 3 による 0 からのカウンタが開始されるのは、常にファイルの保存処理が行われた時点ということになり、このカウンタにより得られる計時値が所定値 T に達した時点でバックアップが実行されるとともに、カウンタが停止させられる。図示の例の場合、時刻 t 9 でカウンタが停止するので、以後、ファイルの新たな保存処理が行われない限り、バックアップ処理が実施されることはない。もちろん、ファイルの新たな保存処理が行われた場合は、再び、計時手段 5 3 による 0 からのカウンタが開始されるので、計時値が所定値 T に達した時点でバックアップが実行される。

【 0 0 4 6 】

なお、図 2 には、説明の便宜上、時刻 t 1 においてファイルを開く処理、時刻 t 8 においてファイルを閉じる処理が実行されることを示したが、本実施形態の場合、ファイルを開いたり閉じたりする処理が、計時手段 5 3 のカウンタ処理やバックアップ実行手段 5 4 のバックアップ処理に影響を及ぼすことはない。これは、ファイルを開いたり、閉じたりする操作は、第 1 の格納場所 2 0 内に保存されているファイルの更新状態には、何ら変化を及ぼさない操作であるため、バックアップのタイミングを考慮する際に、ファイルの開閉操作を考慮する必要はないためである。

【 0 0 4 7 】

ところで、図 2 に示すタイムチャートは、ファイル名「 F 」なる 1 つのファイルに関するバックアップタイミングを示す図であるが、実際には、複数のファイルを同時に開いて作業を進めることも少なくない。このように、複数のファイルを同時に取り扱うためには、上述したデータバックアップ装置 5 0 によって実行される各処理が、個々のファイルごとに別個独立して行われるようにしておけばよい。

【 0 0 4 8 】

具体的には、まず、保存処理検出手段 5 1 には、保存処理が実行された場合に、その処理が、どのファイルについての処理であるのかを検出する機能をもたせておくようにし、計時リセット手段 5 2 には、特定のファイルについての保存処理が行われるたびに、当該特定のファイルについての計時値を 0 にリセットして、計時手段に 0 からカウンタを開始させる機能をもたせておくようにする。

【 0 0 4 9 】

もちろん、計時手段 5 3 には、第 1 の格納場所 2 0 に保存されている個々のファイルごとに、それぞれ別個独立して、保存処理が行われた時点からの経過時間を計時値としてカウンタする機能をもたせておく。図 1 には、計時手段 5 3 内に、 3 つのファイル F a , F

10

20

30

40

50

b, F cのそれぞれについての計時値が別個独立して記録されている例が示されている。具体的には、ファイルF aについての計時値は25分、ファイルF cについての計時値は17分となっており、ファイルF bについてのカウントは停止状態となっている。

【0050】

一方、バックアップ実行手段54には、特定のファイルについての計時値が所定値Tに達したら、当該特定のファイルを第1の格納場所20から第2の格納場所40にバックアップするとともに、当該特定のファイルについての計時値のカウントを停止させる機能をもたせておけばよい。

【0051】

図3は、図1に示すデータバックアップ装置50の具体的な動作手順の一例を示す表である。第1列目は時刻、第2列目はオペレータの処理動作(但し、括弧書きした内容は、データバックアップ装置50によるバックアップ動作)、第3~第5列目は各ファイルF a, F b, F cについての計時値(単位は分:Sはカウント停止状態)を示している。なお、図2に示した所定値Tとして、この例では、T=60分なる設定を行っている。

10

【0052】

はじめに、オペレータの処理動作を簡単に説明しよう。まず、朝8時50分に、このコンピュータに電源を入れ、システムを起動し、8時55分に所定のアプリケーションプログラムを用いて、ファイルF aを開く操作を行っている。以後、9時50分まで、このファイルF aに対する編集操作を行いながら、編集途中のファイルF aを、9時00分および9時10分に保存し、9時50分に編集済みのファイルF aを保存して閉じる操作を行っている。このオペレータの午前中の操作は、以上である。

20

【0053】

続いて、このオペレータは、13時03分に再びファイルF aを開き、13時05分に別なファイルF bを開き、2つのファイルに対する編集操作を同時進行で行っている。そして、13時30分にファイルF bを保存し、13時35分にファイルF aを保存し、14時32分にファイルF aを保存して閉じ、14時55分にファイルF bを保存して閉じている。これで、ファイルF a, F bに対するこの日の作業は完了である。

【0054】

最後に、このオペレータは、15時05分にファイルF cを開き、16時12分にファイルF cを保存して閉じるまで、ファイルF cに対する編集操作を行っている。この間、編集途中のファイルF cを、15時38分および15時52分に保存している。そして、16時31分には、このコンピュータのシステムを終了し、電源を落としている。

30

【0055】

さて、この図3に示されたオペレータの処理動作におけるファイルの保存とは、その時点での更新ファイルを第1の格納場所20に保存する動作であり、オペレータの意志により実行される動作である。これに対して、データバックアップ装置50は、所定のタイミングで、この第1の格納場所20に保存された更新ファイルを、第2の格納場所40にバックアップする処理を行う機能をもっている。このバックアップ処理は、前述したアルゴリズムに基づいて自動的に行われ、オペレータの関知しないところで実行される。

【0056】

40

ここでは、まず、ファイルF aに関するバックアップがどのように行われるかを見てみよう。前述したとおり、ファイルF aに関する計時値は、ファイルF aについての保存処理が実行された時点でリセットされ、このリセット時点をもとして計時値のカウントが実行される。図3に示す例の場合、ファイルF aに関する計時値がリセットされるのは、9時00分、9時10分、9時50分、13時35分、14時32分である。計時値欄に示されている矢印は、その時点で生じる計時値の変遷を示している。たとえば、9時00分の欄に示されている「S 0」は、カウント停止状態Sから計時値0へのリセット操作を示し、9時10分の欄に示されている「10 0」は、計時値10から計時値0へのリセット操作を示している。

【0057】

50

前述したとおり、この例では、所定値  $T = 60$  分なる設定を行っているため、ファイル  $F_a$  についてのバックアップが行われるのは、ファイル  $F_a$  に関する計時値が  $60$  に達することになる  $10$  時  $50$  分ということになる。このとき、バックアップの対象となるファイルは、 $9$  時  $50$  分に保存された更新ファイル  $F_a$  であるので、この更新ファイルのことを、「ファイル  $F_a < 09 : 50 >$  」と記述することにする。結局、 $10$  時  $50$  分に、「ファイル  $F_a < 09 : 50 >$  」のバックアップが実行されることになる。このバックアップの実行により、ファイル  $F_a$  に関するカウントは停止させられる。 $10$  時  $50$  分の欄に示されている「 $60 \quad S$ 」は、このカウント停止の動作を示している。

【 $0058$ 】

こうして、ファイル  $F_a$  に関するカウントは暫く停止した状態が続くが、 $13$  時  $35$  分にファイル  $F_a$  の新たな更新ファイルの保存が行われた時点でリセットが行われ、再び計時値  $0$  からのカウントが再開される。この計時値は、 $14$  時  $32$  分に「 $57$ 」にまで達するものの、再びリセットされ、 $15$  時  $32$  分の時点で、「 $60$ 」に達することになる。したがって、 $15$  時  $32$  分に、「ファイル  $F_a < 14 : 32 >$  」のバックアップ（ $14$  時  $32$  分にファイル  $F_a$  として保存された更新ファイルのバックアップ）が実行される。このバックアップの実行により、ファイル  $F_a$  に関するカウントは再び停止させられ、この日は、もうファイル  $F_a$  についてのバックアップは行われぬ。

【 $0059$ 】

次に、ファイル  $F_b$  に関するバックアップがどのように行われるかを見てみよう。ファイル  $F_b$  に関する計時値は、ファイル  $F_b$  についての保存処理が実行された時点でリセットされ、このリセット時点を  $0$  として計時値のカウントが実行される。図  $3$  に示す例の場合、ファイル  $F_b$  に関する計時値がリセットされるのは、 $13$  時  $30$  分と  $14$  時  $55$  分である。そして、ファイル  $F_b$  についてのバックアップが行われるのは、ファイル  $F_b$  に関する計時値が  $60$  に達することになる  $14$  時  $30$  分と  $15$  時  $55$  分ということになる。具体的には、 $14$  時  $30$  分には、「ファイル  $F_b < 13 : 30 >$  」のバックアップが行われ、 $15$  時  $55$  分には、「ファイル  $F_b < 14 : 55 >$  」のバックアップが行われる。

【 $0060$ 】

このように、ファイル  $F_a$  ,  $F_b$  については、それぞれ適切なタイミングで、バックアップが自動的に実行される。ところが、システム終了時の間に編集作業を実行したファイル  $F_c$  については、このままではバックアップが行われぬ。すなわち、ファイル  $F_c$  に関する計時値は、表に示されているとおり、 $15$  時  $38$  分、 $15$  時  $52$  分、 $16$  時  $12$  分にそれぞれリセットされるため、システム終了時である  $16$  時  $31$  分までには、計時値が「 $60$ 」に達する機会がなく、バックアップ処理のチャンスを失ってしまう。

【 $0061$ 】

このように、システム終了によりバックアップ処理が行われぬ事態を避けるためには、バックアップ実行手段  $54$  に、コンピュータのシステム終了時に、計時手段  $53$  によるカウントが継続して行われている計時継続ファイルを認識し、この計時継続ファイルについて、システム終了直前にバックアップする機能をもたせておくようにすればよい。たとえば、図  $3$  に示す例の場合、システム終了時の  $16$  時  $31$  分の時点において、ファイル  $F_a$  ,  $F_b$  のカウントは停止しているが、ファイル  $F_c$  のカウントはまだ継続していることになる。そこで、システム終了時には、ファイル  $F_c$  を計時継続ファイルとして認識し、これをバックアップする処理が行われるようにすればよい。図示の例の場合、システム終了の直前に、「ファイル  $F_c < 16 : 12 >$  」のバックアップが行われることになる。

【 $0062$ 】

<<< §  $3$  . 本発明の第  $2$  の実施形態 >>>

図  $4$  は、本発明の第  $2$  の実施形態に係るデータバックアップ装置を組み込んだコンピュータシステムの基本構成を示すブロック図である。ここで、メモリ  $10$ 、第  $1$  の格納場所  $20$ 、処理実行部  $30$ 、第  $2$  の格納場所  $40$  は、図  $1$  の第  $1$  の実施形態に示したものと全く同一のものである。一方、データバックアップ装置  $60$  は、本発明の第  $2$  の実施形態に係る装置であり、保存処理検出手段  $61$ 、計時開始手段  $62$ 、計時手段  $63$ 、バックアッ

10

20

30

40

50

プ実行手段64によって構成されている。やはり、これらの各手段も、それぞれコンピュータ上で特定の処理を行う機能要素であり、実際には、コンピュータのハードウェアとソフトウェアの組み合わせによって実現されるものである。

【0063】

この第2の実施形態に係るデータバックアップ装置60も、その役割は、第1の格納場所20内の更新ファイルを、第2の格納場所40へとバックアップする適切なタイミングを決定することにあるが、その基本原理は、前述した第1の実施形態とは若干異なっている。そこで、まず、この基本原理を図5のタイムチャートを参照しながら説明する。このタイムチャートは、図2のタイムチャートと同様に、時間軸の左側に、オペレータの操作タイミングを示し、時間軸の右側に、データバックアップ装置60によるバックアップ処理のタイミングを示すものである。

10

【0064】

オペレータが行った操作は、図2のタイムチャートと全く同様である。すなわち、時刻 $t_1$ において、所定のアプリケーションプログラムを用いて、ファイルを開く操作を行い、以後、当該ファイルに対する編集作業の途中段階の時刻 $t_2$ ,  $t_3$ ,  $t_4$ ,  $t_6$ ,  $t_7$ において、編集途中のファイルを保存する操作を行い、最終的に、時刻 $t_8$ において、当該ファイルを閉じる操作を行っている。なお、この図5のタイムチャートでは、説明の便宜上、時刻 $t_0$ において、コンピュータシステムの起動が行われたものとしている。また、図2のタイムチャートと同様、各更新段階におけるファイルを、それぞれ更新ファイル $F(0) \sim F(5)$ と呼んで相互に区別することにしているが、オペレータから見れば、各更新ファイル $F(0) \sim F(5)$ は、いずれもファイル名「F」で特定される同一のファイルであり、編集作業の途中段階の変遷を示すものに過ぎない。

20

【0065】

この実施形態においても、より効率的なバックアップ処理を実行するために、これらの更新ファイル $F(1) \sim F(5)$ のすべてについてバックアップを行うという方針を採る代わりに、選択されたいくつかについてのみバックアップを行うという方針を採る。この図5の例では、結果的に、図2の例と同様に、ファイル $F(3)$ およびファイル $F(5)$ のみがバックアップの対象となっており、ファイル $F(1)$ ,  $F(2)$ ,  $F(4)$ についてのバックアップは省略されている。2つのファイル $F(3)$ ,  $F(5)$ のみをバックアップの対象として選択する意義は、既に§2で述べたとおりである。

30

【0066】

ここで述べる第2の実施形態の着眼点は、システム起動後、ファイルが初めて保存されたときを起点として、一定時間ごとに、当該ファイルに関する更新ファイル(その時点での最新ファイル)をバックアップすることにより、効率的なバックアップを行う点にある。ここで、「一定時間」をどの程度の時間設定にするかは、実用上の運用形態をみながら適宜設定すればよい。図5に示す実施形態は、この「一定時間」を所定値 $T$ に設定した例である。

【0067】

すなわち、図示の例の場合、システム起動後、ファイル $F$ が初めて保存された時点は、時刻 $t_2$ であるので、この時刻 $t_2$ を起点として、所定値 $T$ に相当する時間間隔ごとにバックアップが実行される。具体的には、時刻 $t_5$ ,  $t_9$ ,  $t_{10}$ というタイミングがバックアップタイミングになる。時刻 $t_5$ では、その時点で最新の更新ファイル $F(3)$ がバックアップされ、時刻 $t_9$ では、その時点で最新の更新ファイル $F(5)$ がバックアップされる。ただ、時刻 $t_{10}$ では、その時点で最新の更新ファイルは依然として $F(5)$ であり、既に時刻 $t_5$ においてバックアップ完了済なので、再度のバックアップは行わない。

40

【0068】

この図5に示すタイムチャートは、ファイル名「F」なる1つのファイルに関するバックアップタイミングを示す図であるが、実際には、複数のファイルを同時に開いて作業を進めた場合にも対応できるように、この図5に示す各処理が、個々のファイルごとに別個

50

独立して行われるようにしておけばよい。図4に示すデータバックアップ装置60は、このような方針で、バックアップのタイミングを決める機能を果たすことになる。以下、このデータバックアップ装置60の各構成要素の機能を説明する。

【0069】

まず、保存処理検出手段61は、コンピュータのメモリ10上に展開されているファイルを第1の格納場所20に所定のファイル名で保存する保存処理が行われたことを検出する構成要素であり、図1に示す保存処理検出手段51と全く同等の機能をもった構成要素である。図5に示すタイムチャートの場合、保存処理検出手段61は、時刻 $t_2$ 、 $t_3$ 、 $t_4$ 、 $t_6$ 、 $t_7$ の各時点において保存処理が行われたことを検出することになる。もちろん、保存処理の検出は、個々のファイルごとに別個独立して行われることになるので、各時点では、どのファイルの保存処理が行われたのかが認識される。

10

【0070】

計時開始手段62および計時手段63は、この保存処理検出手段61の検出結果に基づいて、計時処理を行うための構成要素である。計時開始手段62は、コンピュータのシステム起動後、所定のファイルについて第1の格納場所20への保存処理が初めて行われたときに、当該所定のファイルについて、計時手段63に0からカウントを開始させる機能を有する。計時手段63は、この計時開始手段62から与えられる開始指示に応じて、第1の格納場所20に保存されている個々のファイルごとに、所定時点からの経過時間をそれぞれ計時値としてカウントする。

【0071】

20

一方、バックアップ実行手段64は、特定のファイルについての計時値が所定値Tに達したら、当該計時値を0にリセットして、計時手段63にカウントを継続させるとともに、その時点で第1の格納場所20に保存されている当該特定のファイルについて、バックアップが実行済か否かを認識し、実行済でなかった場合には、当該特定のファイルを第2の格納場所40にバックアップする処理を行う。

【0072】

図5に示すタイムチャートの場合、計時手段63によるカウントは、時刻 $t_2$ から開始するが、計時値が所定値Tに達するたびに、計時値は0にリセットされるので、結局、時間軸上に黒丸で示すタイミングで、計時値のリセットが行われ、0からのカウントが開始されることになる。しかも、この黒丸で示すタイミングごとに、バックアップ実行手段64によって、当該ファイル（その時点で最新の更新ファイル）についてのバックアップが実行済か否かの判断がなされ、実行済でなければバックアップが実行される。

30

【0073】

図5の例では、時刻 $t_5$ における判断では、その時点で最新の更新ファイルF(3)のバックアップはまだ実行されていないので、この時点で更新ファイルF(3)についてのバックアップが実行される。また、時刻 $t_9$ における判断では、その時点で最新の更新ファイルF(5)のバックアップはまだ実行されていないので、この時点で更新ファイルF(5)についてのバックアップが実行される。しかしながら、時刻 $t_{10}$ における判断では、その時点で最新の更新ファイルF(5)のバックアップは既に実行済なので、更新ファイルF(5)について再度のバックアップが実行されることはない。

40

【0074】

なお、特定の更新ファイルについて、バックアップが実行済か否かの判断は、バックアップを実行した際に何らかの記録を残しておくようにすれば、容易に行うことができる。たとえば、ある特定のファイルFについて、バックアップを実行したらフラグ「1」を立て、新たな更新ファイルの保存が行われたら、当該フラグを「0」にする、という操作を行うようにすれば、当該フラグが「1」であれば、最新の更新ファイルについて、バックアップが実行済であることが確認できる。

【0075】

この第2の実施形態では、あるファイルに関してカウントが開始すると、以後、システム終了時まで、カウントが停止することはない。したがって、図5に示す例の場合、時刻

50

t 10以降も、引き続き計時値のカウントが続行され、所定値Tごとに計時値は0にリセットされる。しかしながら、更新ファイルF(5)についてのバックアップは、既に時刻t9において実行済なので、同じ更新ファイルF(5)が再度バックアップされることはない。

【0076】

もっとも、ファイルFについての編集が再開され、新たな更新ファイルが作成され、これが保存された場合は、当該新たな更新ファイルはバックアップの対象になる。図6は、この場合のタイムチャートを示す。すなわち、この図6に示す例では、時刻t9.1において、オペレータは、ファイルF(5)を開いて、ファイルFについての編集を再開し、時刻t9.2において、更新後の新たなファイルF(6)を保存する操作を行っている。この場合、時刻t10における判断では、その時点で最新の更新ファイルは、ファイルF(6)であり、バックアップは実行されていないことになる。したがって、この場合は、時刻t10の時点で、このファイルF(6)についてのバックアップが実行される。

10

【0077】

このように、ここで述べる実施形態の場合、計時手段63によるカウントは、常に所定値Tに達したときに0にリセットされ、その時点での最新の更新ファイルについて、まだバックアップが実行されていない場合には、バックアップが実行されることになる。かくして、この第2の実施形態においても、内容が頻繁に更新されるファイルについて、効率的なバックアップ処理を実行することが可能になる。

【0078】

20

図7は、図4に示すデータバックアップ装置60の具体的な動作手順の一例を示す表である。図3に示す表と同様に、第1列目は時刻、第2列目はオペレータの処理動作(但し、括弧書きした内容は、データバックアップ装置60によるバックアップ動作)、第3~第5列目は各ファイルFa, Fb, Fcについての計時値(単位は分:Sはカウント停止状態)を示している。なお、図5に示した所定値Tとして、この例では、T=60分なる設定を行っている。

【0079】

この図7の表における時刻およびオペレータの処理動作は、図3に示す表と全く同様である。ただ、各ファイルFa, Fb, Fcについての計時値のカウント動作とバックアップ動作は、上述したアルゴリズムに基づいて実行される。以下、この点について説明する。

30

【0080】

まず、ファイルFaに関するバックアップがどのように行われるかを見てみよう。前述したとおり、ファイルFaに関する計時値のカウントは、システム起動後、ファイルFaについての保存処理が初めて実行された時点から開始する。したがって、図示の例では、9時00分から、ファイルFaに関する計時値のカウントが0から開始することになる。9時00分の欄に示されている「S 0」は、カウント停止状態Sから計時値0へのリセット操作を示し、この時点からカウントが開始することを示している。

【0081】

ここで述べる実施形態の場合、ファイルFaの計時値は、所定値T=60に達するたびに0にリセットされる。したがって、図示の例の場合、ファイルFaの計時値は、10時00分、11時00分、12時00分、13時00分、14時00分、15時00分、16時00分、といういわゆる毎正時にリセットされることになる。各毎正時の欄に示されている「60 0」は、計時値が60から0にリセットされることを示している。

40

【0082】

上述したとおり、この計時値が0にリセットされるタイミングで、必要な場合に(最新の更新ファイルが、まだバックアップされていない場合に)、バックアップが実行される。ファイルFaの場合、10時00分のリセット時における最新の更新ファイルは、9時50分に保存された「ファイルFa<09:50>」であり、当該ファイルに対するバックアップはまだ実行されていない。そこで、10時00分において、この「ファイルFa

50

< 09 : 50 >」についてのバックアップが実行される。しかしながら、11時00分、12時00分、13時00分の各リセット時には、その時点における最新の更新ファイルは、依然として9時50分に保存された「ファイルF a < 09 : 50 >」であり、既にバックアップ実行済であるので、これらの時点でのバックアップは行われない。

【0083】

一方、次の14時00分のリセット時における最新の更新ファイルは、13時35分に保存された「ファイルF a < 13 : 35 >」であり、当該ファイルに対するバックアップはまだ実行されていない。そこで、14時00分において、この「ファイルF a < 13 : 35 >」についてのバックアップが実行される。同様に、次の15時00分のリセット時における最新の更新ファイルは、14時32分に保存された「ファイルF a < 14 : 32 >」であり、当該ファイルに対するバックアップはまだ実行されていない。そこで、15時00分において、この「ファイルF a < 14 : 32 >」についてのバックアップが実行される。最後の16時00分のリセット時には、その時点における最新の更新ファイルは、依然として14時32分に保存された「ファイルF a < 14 : 32 >」であり、既にバックアップ実行済であるので、この時点でのバックアップは行われない。

【0084】

次に、ファイルF bに関するバックアップがどのように行われるかを見てみよう。ファイルF bに関する計時値のカウントは、システム起動後、ファイルF bについての保存処理が初めて実行された時点から開始する。したがって、図示の例では、13時30分から、ファイルF bに関する計時値のカウントが0から開始することになる。13時30分の欄に示されている「S 0」は、カウント停止状態Sから計時値0へのリセット操作を示し、この時点からカウントが開始することを示している。

【0085】

ここで述べる実施形態の場合、ファイルF bの計時値は、所定値T = 60に達するたびに0にリセットされる。したがって、図示の例の場合、ファイルF bの計時値は、14時30分、15時30分、16時30分、といういわゆる毎正半時にリセットされることになる。各毎正半時の欄に示されている「60 0」は、計時値が60から0にリセットされることを示している。

【0086】

そして、この計時値が0にリセットされるタイミングで、必要な場合にバックアップが実行される。ファイルF bの場合、14時30分のリセット時における最新の更新ファイルは、13時30分に保存された「ファイルF b < 13 : 30 >」であり、当該ファイルに対するバックアップはまだ実行されていない。そこで、14時30分において、この「ファイルF b < 13 : 30 >」についてのバックアップが実行される。同様に、15時30分のリセット時における最新の更新ファイルは、14時55分に保存された「ファイルF b < 14 : 55 >」であり、当該ファイルに対するバックアップはまだ実行されていない。そこで、15時30分において、この「ファイルF b < 14 : 55 >」についてのバックアップが実行される。しかしながら、最後の16時30分のリセット時には、その時点における最新の更新ファイルは、依然として14時55分に保存された「ファイルF b < 14 : 55 >」であり、既にバックアップ実行済であるので、この時点でのバックアップは行われない。

【0087】

このように、ファイルF a, F bについては、それぞれ適切なタイミングで、バックアップが自動的に実行される。ところが、システム終了時の間に編集作業を実行したファイルF cについては、このままではバックアップが行われない。すなわち、ファイルF cに関する計時値は、表に示されているとおり、システム終了時の16時31分の時点で「53」となっており、まだ所定値「60」には達していないため、バックアップ処理のチャンスを失ってしまう。

【0088】

このように、システム終了によりバックアップ処理が行われない事態を避けるためには

10

20

30

40

50

、 § 2 の第 1 の実施形態で述べた手法と同様に、バックアップ実行手段 6 4 に、コンピュータのシステム終了時に、第 1 の格納場所 2 0 に保存されているファイルのうち、バックアップが実行済でない未バックアップファイルを認識し、この未バックアップファイルについて、システム終了直前にバックアップする機能をもたせておくようにすればよい。たとえば、図 7 に示す例の場合、システム終了時の 1 6 時 3 1 分の時点において、ファイル F a の最新の更新ファイルである「ファイル F a < 1 4 : 3 2 > 」と、ファイル F b の最新の更新ファイルである「ファイル F b < 1 4 : 5 5 > 」についてはバックアップが実行済であるが、ファイル F c の最新の更新ファイルである「ファイル F c < 1 6 : 1 2 > 」についてのバックアップは実行されていない。そこで、システム終了時には、この「ファイル F c < 1 6 : 1 2 > 」を未バックアップファイルとして認識し、これをバックアップする処理が行われるようにすればよい。

10

【 0 0 8 9 】

< < < § 4 . いくつかの変形例 > > >

これまで、本発明に係るデータバックアップ装置を 2 つの実施形態について説明したが、本発明は、これらの実施形態に限定されるものではなく、この他にも種々の態様で実施可能である。ここでは、いくつかの変形例を述べておく。

【 0 0 9 0 】

(1) 第 2 の実施形態の変形例

上述した § 3 で述べた実施形態では、各ファイルについての計時値が所定値 T に達した時点で、これを 0 にリセットする手法を採っているため、図 5 あるいは図 6 のタイムチャートに示すとおり、常に、所定値 T の時間間隔でバックアップをするか否かの判断が行われることになる。

20

【 0 0 9 1 】

これに対して、特定のファイルについての計時値が所定値 T に達したら、当該特定のファイルの最終保存時刻から現在時刻までの経過時間を認識し、この経過時間が所定値 T でなかった場合には、計時値を、0 にリセットする代わりに、当該経過時間に相当する値に設定するような運用も可能である。

【 0 0 9 2 】

図 5 に示す例について、このような運用を行うと、バックアップのタイミングは次のようになる。まず、図示のとおり、時刻 t 5 において、ファイル F についての計時値が所定値 T に達することになる。そこで、この時刻 t 5 において、ファイル F ( 3 ) のバックアップを行う点は、§ 3 で述べた実施形態と全く同様である。ただ、この時点で計時値を、0 にリセットする代わりに、当該ファイル F の最終保存時刻 ( 更新ファイル F ( 3 ) の保存時刻 ) から現在時刻までの経過時間に相当する値に設定するのである。具体的には、図示の例の場合、時刻 t 5 における新たな計時値は、時刻 t 4 ~ t 5 の間の時間に相当する値に設定されることになる。

30

【 0 0 9 3 】

このような計時値設定を行う本旨は、所定値 T までのカウントの起点を、時刻 t 5 ではなく、時刻 t 4 に変更することにある。時刻 t 5 における新たな計時値として、時刻 t 4 ~ t 5 の間の時間に相当する値を設定すれば、次に計時値が所定値 T に達するのは、図示の時刻 t 9 ではなく、それよりも早い時点 ( 時刻 t 4 ~ t 5 の間の時間分だけ早い時点 ) ということになり、別言すれば、時刻 t 4 を起点として所定値 T までのカウントが行われた時点ということになる。

40

【 0 0 9 4 】

このように、§ 3 で述べた実施形態では、バックアップするか否かの判断タイミングは、常に一定周期 T で訪れることになるが、ここで述べる変形例によれば、ファイルの最終保存時刻に応じて、バックアップするか否かの判断タイミングが若干前倒しに設定されることになる。これは、バックアップしたファイルの最終保存時刻から、次の判断タイミングまでの期間が常に一定周期 T になるようにするための配慮である。たとえば、図 5 に示す例の場合、時刻 t 5 で更新ファイル F ( 3 ) のバックアップが行われた後、次のバック

50

アップ判断タイミングは、時刻  $t_9$  となっており、バックアップしたファイルの最終保存時刻から、次の判断タイミングまでの期間は、時刻  $t_4 \sim t_9$  の間の期間となり、所定値  $T$  に比べてかなり長くなっている。本変形例によれば、バックアップしたファイルの最終保存時刻から、次の判断タイミングまでの期間が常に一定周期  $T$  になる。

【0095】

なお、ファイルの最終保存時刻から現在時刻までの経過時間が所定値  $T$  であった場合は、計時値を  $T$  にリセットするのは不都合であるから、通常どおり 0 にリセットすればよい。もちろん、§ 3 で述べた実施形態と、ここで述べた変形例とを比較した場合に、どちらの方が優れているかを一概に結論づけることはできないが、いずれの手法を採っても、本発明の目的が達成できることに変わりはない。

10

【0096】

(2) バックアップ対象の設定

これまでの実施形態では特に触れていなかったが、本発明を実施するにあたっては、第 1 の格納場所 20 に保存されているすべてのファイルを必ずしもバックアップ対象にする必要はない。具体的には、バックアップ対象データを、ボリューム単位、フォルダ単位、もしくはファイル単位で、予め設定できるようにしておくのが好ましい。この場合、計時値のカウントは、バックアップ対象データについてのみ行うようにすれば足りる。

【0097】

また、第 1 の格納場所 20 内の特定のファイルについてバックアップを実行する際、当該ファイル内のデータ全部を第 2 の格納場所 40 に複製する必要はなく、前回のバックアップ時の内容に対する差分データのみを第 2 の格納場所 40 に保存する処理を行うことも可能である。

20

【0098】

(3) 所定値  $T$  の設定

本発明に係るデータバックアップ装置の使い勝手は、計時値と比較する所定値  $T$  をどのような値に設定するかによって大きく左右される。前述の実施形態では、 $T = 60$  分に設定した例を述べたが、この値は、実際のコンピュータシステムにおいて利用するアプリケーションプログラムの種類や、オペレータの作業内容に応じて、適宜設定変更できるようにしておくのが好ましい。もちろん、所定値  $T$  の値は、対象となるファイルごとに異なるように設定してもよい。たとえば、ワープロ用アプリケーションプログラムによって作成されたファイルの場合は  $T = 20$  分、表計算用アプリケーションプログラムによって作成されたファイルの場合は  $T = 10$  分、というように、アプリケーションプログラムごとに設定値を変えることも可能である。あるいは、オペレータの保存操作の頻度をモニターしながら、この頻度に応じて、所定値  $T$  の値を逐次自動変更させるような態様も可能である。

30

【0099】

(4) システム起動 / 終了に関して

本発明を実施するにあたり、システム起動あるいは終了なる文言は、必ずしもコンピュータに対する電源の投入、切断を意味するものではない。本発明におけるシステムの起動状態とは、データバックアップ装置 50, 60 が正常に動作することが可能な状態を意味しており、システムの終了状態とは、データバックアップ装置 50, 60 が正常に動作できない状態を意味している。したがって、いわゆる節電のためのスリープ状態 / スタンバイ状態への移行を、本発明にいうシステム終了として取り扱う実施形態も可能であるし、これらの状態からの復帰を、本発明にいうシステム起動として取り扱う実施形態も可能である。

40

【図面の簡単な説明】

【0100】

【図 1】本発明の第 1 の実施形態に係るデータバックアップ装置を組み込んだコンピュータシステムの基本構成を示すブロック図である。

【図 2】本発明の第 1 の実施形態の基本原理を示すタイムチャートである。

50

【図 3】図 1 に示すデータバックアップ装置 50 の具体的な動作手順の一例を示す表である。

【図 4】本発明の第 2 の実施形態に係るデータバックアップ装置を組み込んだコンピュータシステムの基本構成を示すブロック図である。

【図 5】本発明の第 2 の実施形態の基本原理を示すタイムチャートである。

【図 6】本発明の第 2 の実施形態の基本原理を示す別なタイムチャートである。

【図 7】図 4 に示すデータバックアップ装置 60 の具体的な動作手順の一例を示す表である。

【符号の説明】

【 0 1 0 1 】

1 0 ... メモリ ( R A M )

2 0 ... 第 1 の格納場所 ( 内蔵ハードディスク )

3 0 ... 処理実行部

4 0 ... 第 2 の格納場所 ( バックアップサーバ )

5 0 ... データバックアップ装置

5 1 ... 保存処理検出手段

5 2 ... 計時リセット手段

5 3 ... 計時手段

5 4 ... バックアップ実行手段

6 0 ... データバックアップ装置

6 1 ... 保存処理検出手段

6 2 ... 計時開始手段

6 3 ... 計時手段

6 4 ... バックアップ実行手段

F a , F b , F c ... バックアップ対象ファイル

F ( 0 ) ~ F ( 6 ) ... ファイル F の各時点における更新ファイル

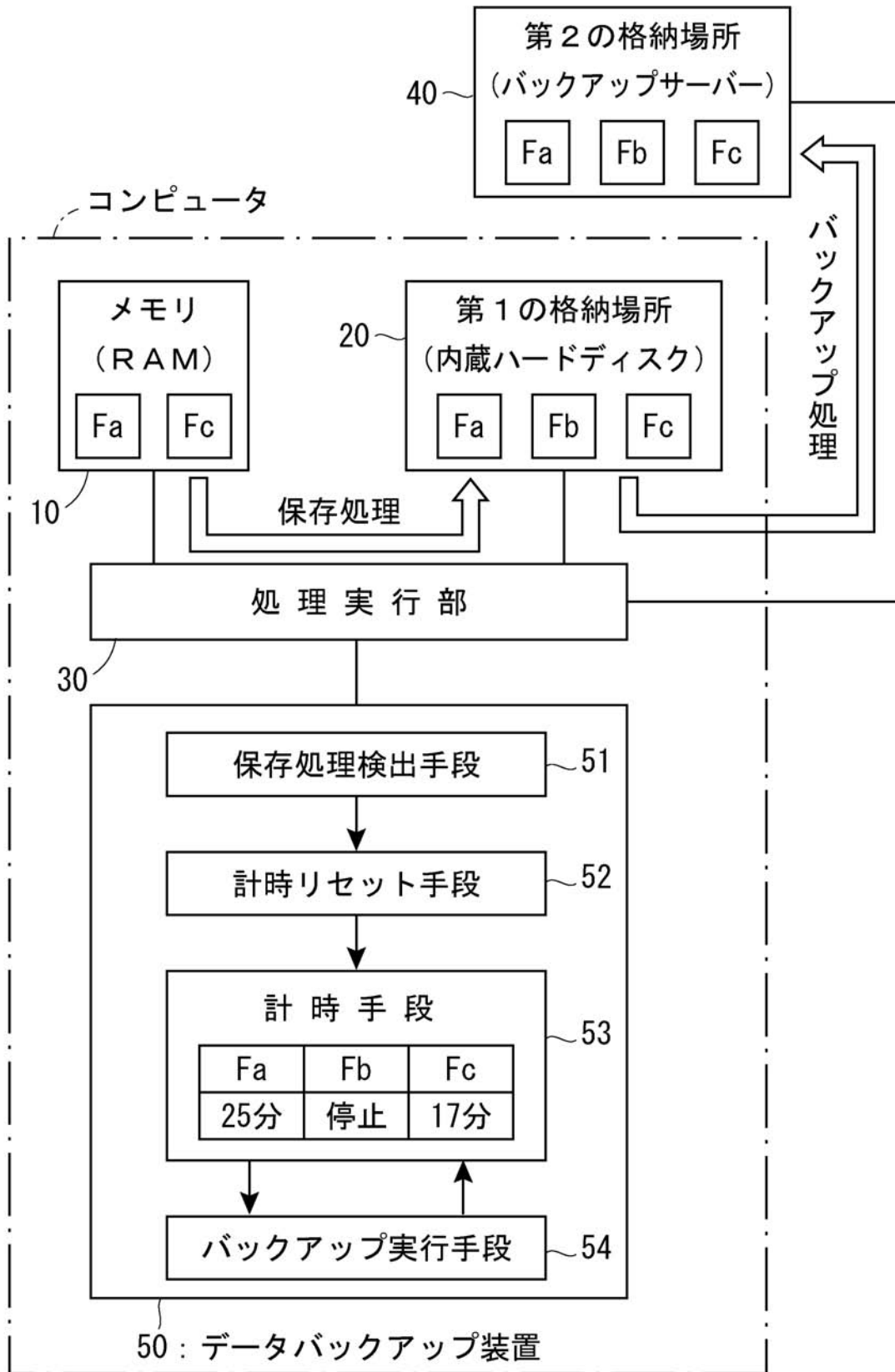
T ... 計時値と比較する値

t 1 ~ t 1 0 ... 各時刻

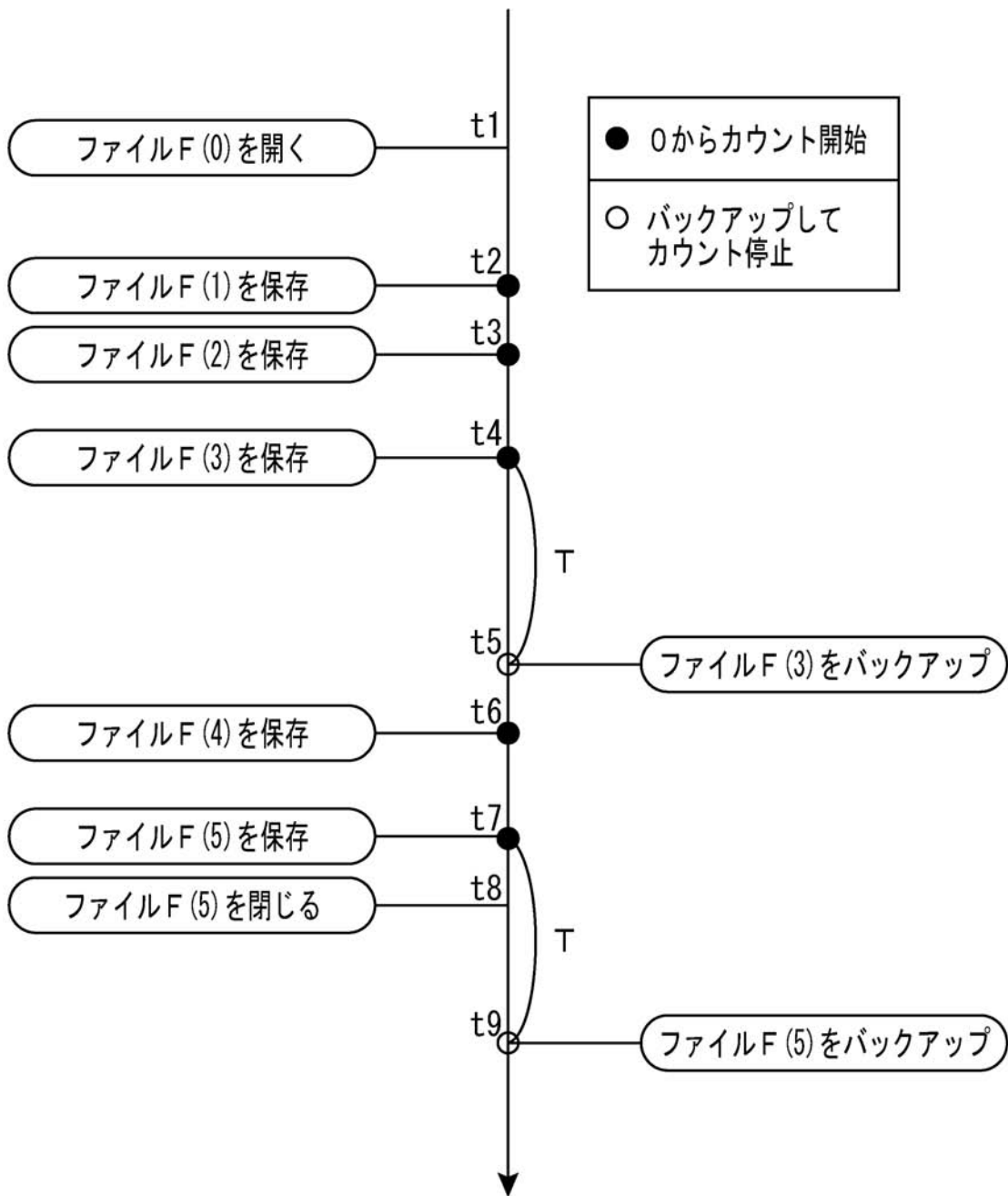
10

20

【図1】



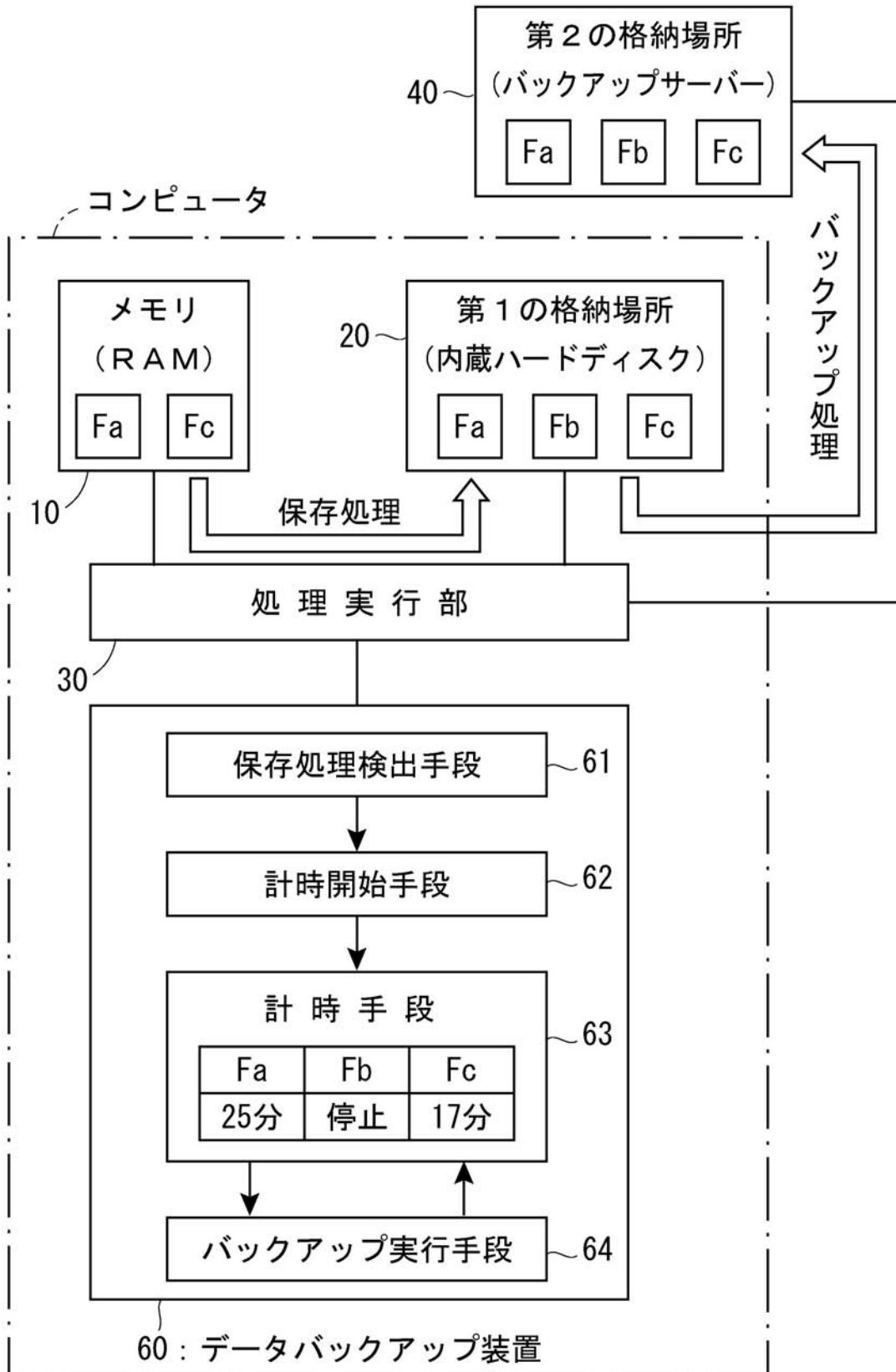
【図2】



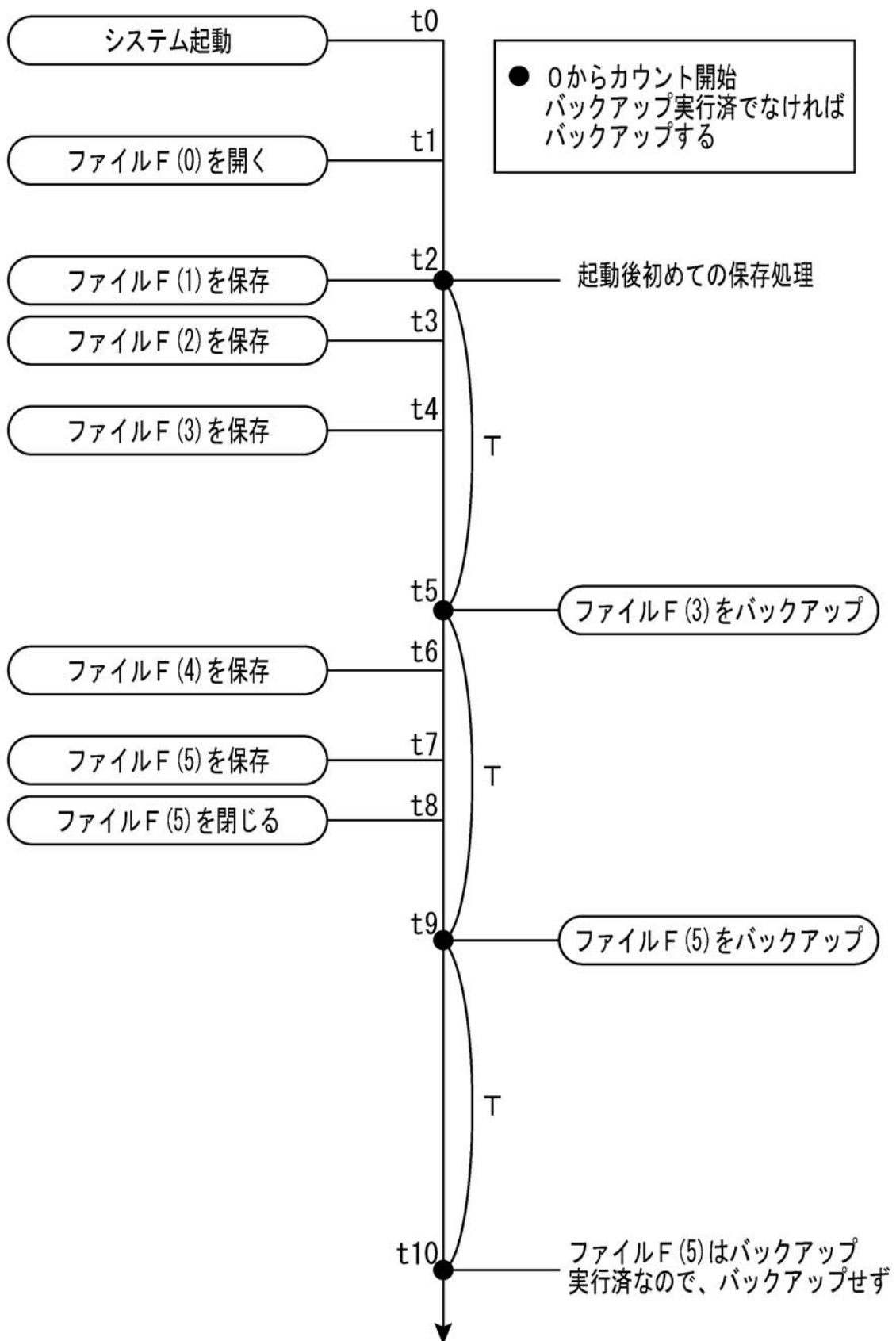
【図3】

時刻	処理動作	計時値		
		Fa	Fb	Fc
08:50	システム起動	S	S	S
08:55	ファイルFaを開く	S	S	S
09:00	ファイルFaを保存	S→0	S	S
09:10	ファイルFaを保存	10→0	S	S
09:50	ファイルFaを保存して閉じる	40→0	S	S
10:00		10	S	S
10:50	(Fa<09:50>のバックアップ実行)	60→S	S	S
11:00		S	S	S
12:00		S	S	S
13:00		S	S	S
13:03	ファイルFaを開く	S	S	S
13:05	ファイルFbを開く	S	S	S
13:30	ファイルFbを保存	S	S→0	S
13:35	ファイルFaを保存	S→0	5	S
14:00		25	30	S
14:30	(Fb<13:30>のバックアップ実行)	55	60→S	S
14:32	ファイルFaを保存して閉じる	57→0	S	S
14:55	ファイルFbを保存して閉じる	23	S→0	S
15:00		28	5	S
15:05	ファイルFcを開く	33	10	S
15:30		58	35	S
15:32	(Fa<14:32>のバックアップ実行)	60→S	37	S
15:38	ファイルFcを保存	S	43	S→0
15:52	ファイルFcを保存	S	57	14→0
15:55	(Fb<14:55>のバックアップ実行)	S	60→S	S
16:00		S	S	S
16:12	ファイルFcを保存して閉じる	S	S	20→0
16:30		S	S	18
16:31	システム終了	S	S	19

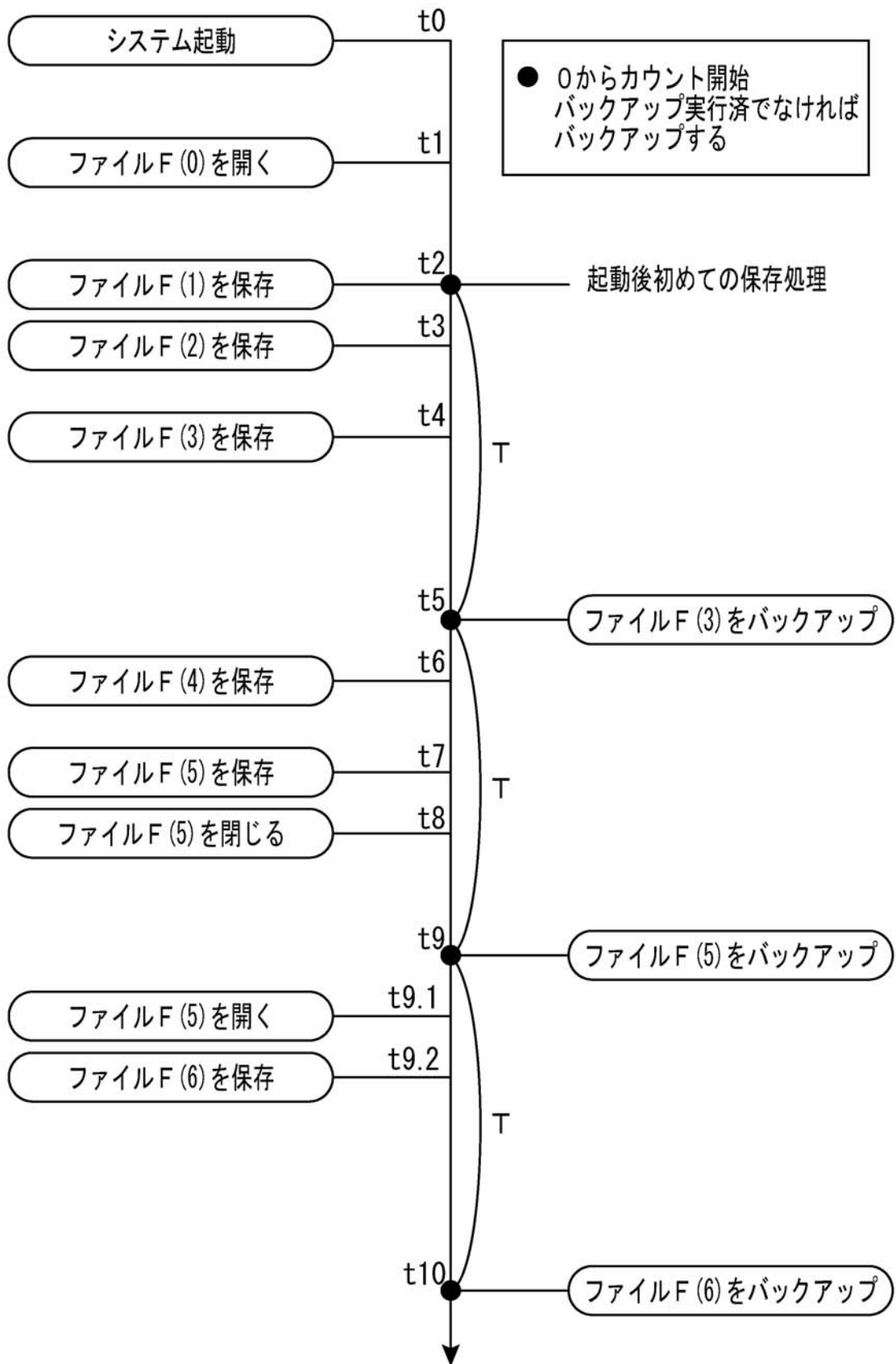
【図4】



【図5】



【図6】



【図7】

時刻	処理動作	計時値		
		Fa	Fb	Fc
08:50	システム起動	S	S	S
08:55	ファイルFaを開く	S	S	S
09:00	ファイルFaを保存	S→0	S	S
09:10	ファイルFaを保存	10	S	S
09:50	ファイルFaを保存して閉じる	50	S	S
10:00	(Fa<09:50>のバックアップ実行)	60→0	S	S
10:50		50	S	S
11:00		60→0	S	S
12:00		60→0	S	S
13:00		60→0	S	S
13:03	ファイルFaを開く	3	S	S
13:05	ファイルFbを開く	5	S	S
13:30	ファイルFbを保存	30	S→0	S
13:35	ファイルFaを保存	35	5	S
14:00	(Fa<13:35>のバックアップ実行)	60→0	30	S
14:30	(Fb<13:30>のバックアップ実行)	30	60→0	S
14:32	ファイルFaを保存して閉じる	32	2	S
14:55	ファイルFbを保存して閉じる	55	25	S
15:00	(Fa<14:32>のバックアップ実行)	60→0	30	S
15:05	ファイルFcを開く	5	35	S
15:30	(Fb<14:55>のバックアップ実行)	30	60→0	S
15:32		32	2	S
15:38	ファイルFcを保存	38	8	S→0
15:52	ファイルFcを保存	52	22	14
15:55		55	25	17
16:00		60→0	30	22
16:12	ファイルFcを保存して閉じる	12	42	34
16:30		30	60→0	52
16:31	システム終了	31	1	53

---

フロントページの続き

- (72)発明者 吉川 和寿  
東京都新宿区市谷加賀町一丁目1番1号 大日本印刷株式会社内
- (72)発明者 半田 富己男  
東京都新宿区市谷加賀町一丁目1番1号 大日本印刷株式会社内

審査官 北村 学

- (56)参考文献 特開平06-309210(JP,A)  
特開平06-250902(JP,A)  
特開平11-053240(JP,A)  
特開平04-253235(JP,A)  
特開平06-044118(JP,A)  
特開平06-083685(JP,A)  
特開平09-198289(JP,A)  
特開平11-134234(JP,A)  
特開2001-014194(JP,A)  
特開2002-091752(JP,A)  
特開2002-334002(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06F 12/00  
G06F 3/06