(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3657740号
(P3657740)
(45)発行日　平成17年6月8日(2005.6.8)　(24)登録日　平成17年3月18日(2005.3.18)

(51)Int.Cl.[7]	F I		
G06K 17/00	G06K	17/00	D
G06F 9/06	G06F	9/06	550Z
G06F 12/14	G06F	12/14	320A
G06K 19/07	G06K	19/00	N

請求項の数 5　(全 14 頁)

(21)出願番号　特願平9-135594
(22)出願日　平成9年5月26日(1997.5.26)
(65)公開番号　特開平10-326329
(43)公開日　平成10年12月8日(1998.12.8)
審査請求日　平成13年1月24日(2001.1.24)

(73)特許権者　000006013
三菱電機株式会社
東京都千代田区丸の内二丁目２番３号
(74)代理人　100089118
弁理士　酒井　宏明
(72)発明者　石原　智史
東京都千代田区丸の内二丁目２番３号　三菱電機株式会社内

審査官　丹治　彰

最終頁に続く

(54)【発明の名称】ＩＣカードシステム

(57)【特許請求の範囲】

【請求項１】

ソフトウェアが登録されたＩＣカードと、該ＩＣカードとの間でデータを授受するコントローラとを含むＩＣカードシステムであって、

前記ＩＣカードは、

ＩＣカードのソフトウェアとそのソフトウェアを実行するのに必要なアーキテクチャ情報とを別の領域に格納するＩＣカードメモリと、

ソフトウェアを独立して実行するとともに各部を制御するＩＣカード制御部と、

前記コントローラとの間でデータを送受信する通信用ドライバとを備え、

前記コントローラは、

ソフトウェアを実行するとともに各部を制御するコントローラ制御部と、

前記ＩＣカードとの間でデータを送受信する通信用ドライバとを備え、

前記ＩＣカードメモリから前記コントローラ制御部がアーキテクチャ情報を受け取ってコントローラ上で実行可能なソフトウェアか否かを判断し、実行可能な場合には、前記ＩＣカードメモリからソフトウェアを転送させてコントローラ上で実行し、実行不可能な場合には、前記ＩＣカード上でソフトウェアを実行することを特徴とするＩＣカードシステム。

【請求項２】

ソフトウェアが登録されたＩＣカードと、該ＩＣカードとの間でデータを授受するコントローラとを含むＩＣカードシステムであって、

前記コントローラは、
コントローラのソフトウェアとそのソフトウェアを実行するのに必要なアーキテクチャ情報とを別の領域に格納するコントローラメモリと、
ソフトウェアを実行するとともに各部を制御するコントローラ制御部と、
前記ＩＣカードとの間でデータを送受信する通信用ドライバとを備え、
前記ＩＣカードは、
ソフトウェアを独立して実行するとともに各部を制御するＩＣカード制御部と、
前記コントローラとの間でデータを送受信する通信用ドライバとを備え、
前記コントローラメモリから前記ＩＣカード制御部がアーキテクチャ情報を受け取ってＩＣカード上で実行可能なソフトウェアか否かを判断し、実行可能な場合には、前記コントローラメモリからソフトウェアを転送させてＩＣカード上で実行し、実行不可能な場合には、前記コントローラ上でソフトウェアを実行することを特徴とするＩＣカードシステム。

【請求項３】
前記ＩＣカードと前記コントローラとの間でソフトウェアを転送する際に、当該ソフトウェアのインストール可能回数を設定し、その設定回数以上に行われるインストールをソフト上で禁止するインストールカウンタ機能を持たせたことを特徴とする請求項１または２に記載のＩＣカードシステム。

【請求項４】
前記ＩＣカードメモリから前記コントローラにインストールしたソフトウェアをアンインストールする場合は、前記ＩＣカードメモリのインストールカウンタのカウント値をインクリメントすることを特徴とする請求項３に記載のＩＣカードシステム。

【請求項５】
前記ＩＣカードメモリに格納されるソフトウェアが複数あって、各ソフトウェア毎の設定を可能としたことを特徴とする請求項３に記載のＩＣカードシステム。

【発明の詳細な説明】

【０００１】

【発明の属する技術分野】
本発明は、ＩＣカードシステムに係り、さらに詳しくは、複製または閲覧が制限されるセキュリティの要求されるソフトウェアが登録されたＩＣカードと、そのＩＣカードとの間でデータの授受を行うコントローラとを含むＩＣカードシステムに関する。

【０００２】

【従来の技術】
近年、ソフトウェアの配布は、フロッピー（Ｒ）ディスクやＣＤ－ＲＯＭなどの記憶媒体を用いて行われている。これらの記憶媒体は、コントローラやパーソナルコンピュータ上で容易に複製することが可能であるため、配布数量や条件のある配布を行うには記憶媒体だけではなく、(1)インストール時にパスワードを必要とする方式、(2)ネットワーク上で同時に実行されているソフトウェアを監視する方式、さらに、厳密さが要求される場合には、(3)実行時にパスワードを格納した付加回路を必要とする方式などがあった。

【０００３】
また、様々な種類のコントローラやパーソナルコンピュータのために同一の機能を持つソフトウェアを配布する場合は、コントローラやパーソナルコンピュータの機種やＣＰＵ、オペレーティングシステムに応じたソフトウェアを個別に用意する必要があった。

【０００４】
例えば、従来のソフトウェアは、図８に示されるように、フロッピー（Ｒ）ディスク８０のソフトウェア領域８２に格納されており、コントローラ９０のフロッピー（Ｒ）ディスク用ドライバ９２を介して、ソフトウェアの読み出しや書き込み等のデータ交換を行うことができる。フロッピー（Ｒ）ディスク８０から読み出されたソフトウェアは、メモリ９４や外部記憶部９８に格納され、コントローラ９０のＣＰＵ９６によって実行されていた。

【０００５】
また、不図示のＣＤ－ＲＯＭ等の記憶媒体に格納されているソフトウェアの場合も上記と同様であり、ＣＤ－ＲＯＭドライバが設けられたパーソナルコンピュータによってＣＤ－ＲＯＭからソフトウェアを読み出して、パーソナルコンピュータによりそのソフトウェアを実行していた。
【０００６】
このため、ソフトウェアの開発者は、各アーキテクチャに対応したソフトウェアをそれぞれ開発し、それぞれのアーキテクチャ用のソフトウェアとして個別に配布していた。
【０００７】
【発明が解決しようとする課題】
しかしながら、このような従来のソフトウェアの配布方式にあっては、上述した(1)の配布数量や条件のあるソフトウェアを管理するためにパスワードを利用する方式の場合、パスワードが適正に利用されていないと適正な配布数量や条件のある配布が行えなくなるという不都合があった。
【０００８】
また、上述した(3)において、ソフトウェアの配布数量や条件に特に厳密さが要求される場合は、実行時にパスワードを格納した付加回路を必要とするが、特殊な付加回路のため拡張スロットを使用しなくてはならないという不都合があった。もっとも、ソフトウェアは、コントローラやパーソナルコンピュータに読み込んで利用するものである以上、基本的にソフトウェアの閲覧そのものを制限することは事実上不可能であった。
【０００９】
さらに、ソフトウェアの開発は、上述したように異なるアーキテクチャ毎に開発する必要があるため、対応するアーキテクチャの数だけソフトウェアを開発しなければならず、複数のアーキテクチャに対応したソフトウェアの開発は長期間を要し、面倒なものになるという不都合があった。
【００１０】
本発明は、かかる従来技術の有する不都合に鑑みてなされたものであり、複製または閲覧が制限される厳密なセキュリティの要求されるソフトウェア、あるいは様々な配布条件のソフトウェアをＩＣカードを用いて容易に自動管理することができるＩＣカードシステムを得ることを第１の目的とする。
【００１１】
また、様々な制御部を用いてソフトウェアを実行することが可能なＩＣカードシステムを得ることを第２の目的としている。
【００１７】
【課題を解決するための手段】
上記の目的を達成するために、この発明に係るＩＣカードシステムにあっては、ソフトウェアが登録されたＩＣカードと、該ＩＣカードとの間でデータを授受するコントローラとを含むＩＣカードシステムであって、前記ＩＣカードは、ＩＣカードのソフトウェアとそのソフトウェアを実行するのに必要なアーキテクチャ情報とを別の領域に格納するＩＣカードメモリと、ソフトウェアを独立して実行するとともに各部を制御するＩＣカード制御部と、前記コントローラとの間でデータを送受信する通信用ドライバとを備え、前記コントローラは、ソフトウェアを実行するとともに各部を制御するコントローラ制御部と、前記ＩＣカードとの間でデータを送受信する通信用ドライバとを備え、前記ＩＣカードメモリから前記コントローラ制御部がアーキテクチャ情報を受け取ってコントローラ上で実行可能なソフトウェアか否かを判断し、実行可能な場合には、前記ＩＣカードメモリからソフトウェアを転送させてコントローラ上で実行し、実行不可能な場合には、前記ＩＣカード上でソフトウェアを実行するものである。
【００１８】
これによれば、コントローラは、ＩＣカードメモリからソフトウェアを実行するのに必要な資源情報であるアーキテクチャ情報を通信によって受け取り、これに基づいてコント

ローラ側でのソフトウェアの実行の可否を判断し、ソフトウェアがコントローラ側で実行可能な場合は、ＩＣカードメモリからソフトウェアを転送させて実行し、実行不可能な場合は、ＩＣカード上でソフトウェアを実行するようにする。このため、ソフトウェアは、必ずしもコントローラ側の制御部に依存しないソフトウェアで作成することができるため、自由度が大きく、さらにＩＣカードとコントローラとの間で資源情報が一致していれば、ＩＣカードのソフトウェアをコントローラ側で実行することもできる。

【００１９】

つぎの発明に係るＩＣカードシステムにあっては、ソフトウェアが登録されたＩＣカードと、該ＩＣカードとの間でデータを授受するコントローラとを含むＩＣカードシステムであって、前記コントローラは、コントローラのソフトウェアとそのソフトウェアを実行するのに必要なアーキテクチャ情報とを別の領域に格納するコントローラメモリと、ソフトウェアを実行するとともに各部を制御するコントローラ制御部と、前記ＩＣカードとの間でデータを送受信する通信用ドライバとを備え、前記ＩＣカードは、ソフトウェアを独立して実行するとともに各部を制御するＩＣカード制御部と、前記コントローラとの間でデータを送受信する通信用ドライバとを備え、前記コントローラメモリから前記ＩＣカード制御部がアーキテクチャ情報を受け取ってＩＣカード上で実行可能なソフトウェアか否かを判断し、実行可能な場合には、前記コントローラメモリからソフトウェアを転送させてＩＣカード上で実行し、実行不可能な場合には、前記コントローラ上でソフトウェアを実行するものである。

【００２０】

これによれば、ＩＣカード制御部は、コントローラと通信してコントローラメモリ内のアーキテクチャ情報を受け取り、ＩＣカード側でコントローラ上のソフトウェアが実行可能か否かを判断し、実行可能な場合はコントローラ上のソフトウェアをＩＣカードに転送して実行することができる。このように、ＩＣカード制御部でコントローラ上のソフトウェアを実行することができるため、コントローラとＩＣカード内で異なるソフトウェアをそれぞれ実行させて並列処理することも可能となり、処理速度を向上させることができる。

【００２１】

つぎの発明に係るＩＣカードシステムにあっては、前記ＩＣカードと前記コントローラとの間でソフトウェアを転送する際に、当該ソフトウェアのインストール可能回数を設定し、その設定回数以上に行われるインストールをソフト上で禁止するインストールカウンタ機能を持たせたものである。

【００２２】

これによれば、ソフトウェアのインストール可能回数をソフト上で設定するインストールカウンタ機能を備えているため、ソフトウェアの配布前にインストール可能回数を設定することによりインストール可能なソフトウェアの場合は、一旦コントローラ側にインストールすればＩＣカードを装着しなくてもソフトウェアを実行することが可能となる。また、インストールカウンタに設定された数以上のインストールを禁止することができるため、ソフトウェアの無制限なコピーを防止し、厳密なソフトウェアの配布本数制限が可能となる。さらに、インストール可能回数の設定は、パスワードを用いて行うため、ソフトウェアの配布本数制限をさらに厳密なものとすることができる。

【００２３】

つぎの発明に係るＩＣカードシステムにあっては、前記ＩＣカードメモリから前記コントローラにインストールしたソフトウェアをアンインストールする場合は、前記ＩＣカードメモリのインストールカウンタのカウント値をインクリメントするものである。

【００２４】

これによれば、ＩＣカードからコントローラにソフトウェアをインストールしても、そのインストールされたソフトウェアをコントローラから戻した場合は（アンインストール）、ＩＣカードメモリのインストールカウンタのカウント値がインクリメントされるため、再び別のコントローラにインストールすることが可能となる。このように、ソフトウェ

アの配布本数制限をインストールカウンタにより厳密に管理することができる。
【００２５】
つぎの発明に係るＩＣカードシステムにあっては、前記ＩＣカードメモリに格納されるソフトウェアが複数あって、各ソフトウェア毎の設定を可能としたものである。
【００２６】
これによれば、ＩＣカードメモリに複数のソフトウェアが格納されて配布可能であり、それらの各ソフトウェア毎にインストールカウンタの設定や実行許可フラグを設定することができるため、様々な属性を持ったソフトウェアを一枚のＩＣカードで同時に配布することが可能となる。
【００２９】
【発明の実施の形態】
以下、この発明に係るＩＣカードシステムの実施の形態を図面に基づいて詳細に説明する。
【００３０】
（実施の形態１）
図１には、実施の形態１に係るＩＣカードシステムのＩＣカード１０の概略構成図が示され、図２には、図１のアーキテクチャ情報領域２０に格納されている情報例が示され、図３には、図１のＩＣカード１０とこれが装着されるコントローラ５０とで構成されるＩＣカードシステムの概略構成図が示されている。
【００３１】
この実施の形態１に係るＩＣカードシステムは、配布したいソフトウェアを記憶媒体であるＩＣカード１０に格納しておき、コントローラ５０にＩＣカード１０を装着することにより、ソフトウェアをインストールしたり、装着したままソフトウェアを実行させたりするものである。
【００３２】
まず、ＩＣカード１０は、図１において、ソフトウェアを実行するＩＣカード制御部としてのＣＰＵ（Central Processing Unit）１２と、ソフトウェアやそのソフトウェアを実行するのに必要な資源情報としてのアーキテクチャ情報等が格納されるＩＣカードメモリとしてのメモリ１４と、後述するコントローラとの間でデータのやり取りする際に使う通信用メモリ１６とを備えている。
【００３３】
メモリ１４内は、さらにソフトウェアが格納されるソフトウェア領域１８と、ＩＣカード１０の構成を表すアーキテクチャ情報が格納されるアーキテクチャ情報領域２０と、通信用のドライバプログラムが格納されている通信用ドライバとしての通信用ドライバ領域２２とに分けられる。
【００３４】
また、上述したソフトウェア領域１８は、さらにソフトウェアプログラムが格納されるプログラム領域２４と、そのプログラム領域２４に格納されたソフトウェアのシリアルナンバーを格納するシリアルナンバー領域２６と、ソフトウェアのインストール可能回数を設定するインストールカウンタ領域２８、ソフトウェアの実行の許可あるいは不許可を表す実行許可フラグが格納される実行フラグ領域３０とに分けられる。なお、インストールカウンタ領域２８のインストール可能回数と実行フラグ領域３０の実行許可フラグの設定時、あるいは設定変更時には所定のパスワードを必要とすることにより、配布条件やソフトウェアの誤使用などを厳密に管理するようにしたものである。
【００３５】
上述したアーキテクチャ情報領域２０に格納されるアーキテクチャ情報とは、ソフトウェアの実行に必要な資源の情報であり、ここでは、図２に示されるように、ＣＰＵの種類４０、メモリ容量４２、ＣＰＵ負荷４４等の情報などから構成されている。
【００３６】
図３に示されるＩＣカード１０は、図１と同じであるため構成説明を省略し、そのＩＣ

カード１０が装着されるコントローラ５０の構成について説明する。図３において、コントローラ５０は、ＩＣカード１０との間で通信を行う通信用ドライバとしてのＩＣカード用ドライバ５２と、コントローラ５０内で実行されるソフトウェアや各種パラメータ等が格納されるコントローラメモリとしてのメモリ５４と、コントローラ５０の各部を制御するコントローラ制御部としてのＣＰＵ５６と、メモリ５４に接続された外部記憶部５８等を備えており、上記メモリ５４内には、コントローラ５０上のソフトウェアの実行に必要な資源の情報としてのアーキテクチャ情報を格納するアーキテクチャ情報領域６０が設けられている。

【００３７】

つぎに、図１ないし図３におけるＩＣカード１０から配布されたソフトウェアの実行動作について、図４のフローチャートを用いて簡単に説明する。まず、図３に示されるＩＣカード１０をコントローラ５０に装着する（ステップＳ１００）。

【００３８】

そして、コントローラ５０のＩＣカード用ドライバ５２は、ＩＣカード１０が装着されたことを検出すると、ＩＣカード１０上のメモリ１４内の通信用ドライバ領域２２と通信用メモリ１６（図１参照）とを用いてＩＣカード１０との間で通信を開始する（ステップＳ１０２）。

【００３９】

ＩＣカード１０は、コントローラ５０から実行開始信号が受信されるのを待機している（ステップＳ１０４）。これは、実行開始信号が受信されない間はＮＯとなり、再びステップＳ１０４に戻って実行開始信号が受信されるまで繰り返し判断される。

【００４０】

ステップＳ１０４で実行開始信号が受信されると、ソフトウェアの実行開始信号がコントローラ５０からＩＣカード用ドライバ５２を経由して、ＩＣカード１０上に転送されて通知される（ステップＳ１０６）。

【００４１】

ＩＣカード１０は、コントローラ５０からソフトウェアの実行開始信号を受信すると、メモリ１４内に格納されていたソフトウェアをＣＰＵ１２により実行が開始する（ステップＳ１０８）。

【００４２】

このようにして、ソフトウェアを自ら実行したＩＣカード１０は、その実行結果だけをコントローラ５０のＩＣカード用ドライバ５２を経由して出力する。

【００４３】

以上説明したように、この実施の形態１によると、コントローラ５０とＩＣカード１０との間の全ての通信は、通信用メモリ１６を通して行われるが、上述したように、ソフトウェアの出力結果以外の全ての処理はＩＣカード１０の内部で行われるため、ＩＣカード１０のメモリ１４に格納されているソフトウェア（プログラム）の内容を隠蔽することが可能となり、完全なセキュリティを守ることができるとともに、ソフトウェアの利用も可能である。

【００４４】

また、この実行されるソフトウェアは、ＩＣカード１０内のＣＰＵ１２で実行されるため、コントローラ５０側のＣＰＵ５６とアーキテクチャが異なっていても、これに依存しないソフトウェアで作成することが可能である。

【００４５】

さらに、ＩＣカード１０内のＣＰＵ１２でソフトウェアが実行されるため、コントローラ５０内で他のソフトウェアを実行しても、これに関わらず安定した実行速度を確保することができるとともに、並列処理することにより処理速度の向上が図れる。

【００４６】

（実施の形態２）

本発明の実施の形態２を図１ないし図３の構成図、および図５のフローチャートに基づ

いて説明する。ここで、前述した実施の形態１と同一若しくは同等の構成部分については、同一の符号を付すとともにその説明を簡略化し若しくは省略するものとする。

【００４７】

この実施の形態２における特徴的な構成は、図３に示されるように、ＩＣカード１０のメモリ１４内にソフトウェアを実行するのに必要な資源情報としてのアーキテクチャ情報が格納されたアーキテクチャ情報領域２０を有しており、コントローラ５０側には、ＣＰＵ５６を有している。

【００４８】

このＣＰＵ５６は、ＩＣカード１０との間で通信を行ってＩＣカード１０側のアーキテクチャ情報を受け取り、コントローラ５０側で実行可能なソフトウェアか否かを判断し、実行可能な場合はＩＣカード１０からソフトウェアを転送させて、コントローラ５０側で転送したソフトウェアを実行するものである。

【００４９】

つぎに、実施の形態２に係るＩＣカード１０により配布されたソフトウェアをコントローラ５０上のＣＰＵ５６で実行する動作を図５に示すフローチャートを用いて説明する。

【００５０】

まず、図３に示されるように、ＩＣカード１０をコントローラ５０に装着すると（ステップＳ１１０）、コントローラ５０のＩＣカード用ドライバ５２がＩＣカード１０の装着を検出し、ＩＣカード１０上の通信用ドライバ（図１の通信用ドライバ領域２２）と通信用メモリとを介して通信を開始する（ステップＳ１１２）。

【００５１】

コントローラ５０のＩＣカード用ドライバ５２は、ＩＣカード１０のアーキテクチャ情報領域２０内のアーキテクチャ情報９を読み取って（ステップＳ１１４）、つぎのステップＳ１１６に移行する。ステップＳ１１６では、コントローラ５０のＣＰＵ５６は、ＩＣカード１０のアーキテクチャ情報であるＣＰＵの種類、メモリ容量、ＣＰＵ負荷に基づいてＩＣカード１０上のソフトウェアがコントローラ５０上で実行可能か否かを、コントローラ５０側のメモリ５４内のアーキテクチャ情報領域６０のアーキテクチャ情報と比較して判断する。

【００５２】

ステップＳ１１６で実行不可能と判断された場合は、ＩＣカード１０上でそのソフトウェアが実行される（ステップＳ１１８）。また、実行可能と判断された場合は、ＩＣカード１０上のＣＰＵ１２によりソフトウェアを通信用メモリ１６にコピーして、コントローラ５０のＩＣカード用ドライバ５２を経由してコントローラ５０上のメモリ５４にソフトウェアが転送されてロードされる（ステップＳ１２０）。この転送されたソフトウェアは、コントローラ５０上のＣＰＵ５６で実行される。

【００５３】

以上述べたように、実施の形態２では、ソフトウェアは、コントローラ側で実行可能か否かを判断して、実行可能であればコントローラ側に転送して実行することができ、また、ソフトウェアの実行がコントローラ側で不可能であっても、ＩＣカード側で実行すれば良いため、必ずしもコントローラ側の制御部に依存しないソフトウェアで作成することが可能であるため、ソフトウェアの自由度が高くなる。なお、本実施の形態２では、一般に処理能力が高いコントローラ側でアーキテクチャ情報に基づいて実行可能か否かを判断したが、ＩＣカードがコントローラのアーキテクチャ情報を収集し、コントローラで実行可能な場合のみ転送するようにしてもよい。

【００５４】

（実施の形態３）

本発明の実施の形態３を図１ないし図３の構成図、および図６のフローチャートに基づいて説明する。ここで、前述した実施の形態１と同一若しくは同等の構成部分については、同一の符号を付すとともにその説明を簡略化し若しくは省略するものとする。

【００５５】

この実施の形態３における特徴的な構成は、図３に示されるように、コントローラ５０のメモリ５４内にコントローラ上のソフトウェアを実行するのに必要な資源情報としてのアーキテクチャ情報が格納されたアーキテクチャ情報領域６０を有しているとともに、ＩＣカード１０との間で通信を行うＩＣカード用ドライバ５２を有している。

【００５６】

ＩＣカード１０側のＣＰＵ１２は、コントローラ５０との間で通信を行ってコントローラ５０側のアーキテクチャ情報（アーキテクチャ情報領域６０）を受け取り、ＩＣカード１０側で実行可能なソフトウェアか否かを判断し、実行可能な場合はコントローラ５０からソフトウェアを転送して、ＩＣカード１０側で転送したソフトウェアを実行するものである。

【００５７】

つぎに、実施の形態３に係るコントローラ５０上のソフトウェアをＩＣカード１０上のＣＰＵ１２で実行する動作を図６に示すフローチャートを用いて説明する。

【００５８】

まず、図３に示されるように、ＩＣカード１０をコントローラ５０に装着すると（ステップＳ１３０）、コントローラ５０のＩＣカード用ドライバ５２がＩＣカード１０上の通信用ドライバ（図１の通信用ドライバ領域２２）と通信用メモリとを介して通信を開始する（ステップＳ１３２）。

【００５９】

ＩＣカード側のＣＰＵ１２は、最初にコントローラ５０のメモリ５４内のアーキテクチャ情報領域６０のアーキテクチャ情報を読み取って（ステップＳ１３４）、つぎのステップＳ１３６に移行する。ステップＳ１３６では、ＩＣカード１０のＣＰＵ１２は、ＩＣカード１０のアーキテクチャ情報であるＣＰＵの種類、メモリ容量、ＣＰＵ負荷と比較して、コントローラ５０上のソフトウェアがＩＣカード１０上で実行可能か否かを判断する。

【００６０】

ステップＳ１３６で実行不可能と判断された場合は、コントローラ５０上でそのソフトウェアが実行される（ステップＳ１３８）。また、実行可能と判断された場合、ＣＰＵ１２は、コントローラ５０側のＩＣカード用ドライバ５２を経由して、コントローラ上のソフトウェアをＩＣカード１０上の通信用メモリ１６に転送し、さらにその通信用メモリ１６からソフトウェアはメモリ１４に転送されてロードされる（ステップＳ１４０）。この転送されたソフトウェアは、ＩＣカード１０上のＣＰＵ１２で実行される（ステップＳ１４２）。

【００６１】

以上述べたように、実施の形態３では、ＩＣカード１０上のＣＰＵ１２でコントローラ上のソフトウェアを実行することができるため、コントローラ５０とＩＣカード１０内で異なるソフトウェアをそれぞれ実行させて並列処理することも可能となり、処理速度を向上させることが可能である。

【００６２】

（実施の形態４）

本発明の実施の形態４を図１ないし図３の構成図に基づいて説明する。ここで、前述した実施の形態１と同一若しくは同等の構成部分については、同一の符号を付すとともにその説明を簡略化し若しくは省略するものとする。

【００６３】

この実施の形態４では、図１に示されるように、ＩＣカード１０のメモリ１４のソフトウェア領域１８内のインストールカウンタ領域２８について説明する。まず、ＩＣカード１０のメモリ１４中に保存されているソフトウェア領域１８には、インストールカウンタが保持されている。プログラム領域２４に格納されているソフトウェアがインストール可能なソフトウェアの場合は、インストールカウンタ領域２８内にインストール可能な数を予め設定することができる。

【００６４】

例えば、インストールカウンタが１以上に設定されている場合は、インストールが可能となる。プログラム領域２４に格納されているソフトウェアとシリアルナンバー領域２６に保持されているソフトウェアのシリアルナンバーは、図３に示されるＩＣカード用ドライバ５２を経由してコントローラ５０側のメモリ５４に転送されて、外部記憶部５８に保存される。そして、ソフトウェアがインストールされた場合は、ＩＣカード１０のインストールカウンタが１減らされることになる。

【００６５】

このようにしたため、外部記憶部５８にインストールされたソフトウェアは、ＩＣカード１０を外してもソフトウェアを実行することが可能である。このため、通常のＦＤやＣＤ－ＲＯＭのような記憶媒体と同様の使用方法が可能であると同時に、ソフトウェアのインストール数を厳密に管理することが可能になる。

【００６６】

また、この実施の形態４では、ＩＣカード１０からインストールできるソフトウェアにおけるインストール可能な数が、インストールカウンタに登録されている。このため、ＩＣカード１０からコントローラ５０上にインストールされたソフトウェアが不要になった場合に、ＩＣカード１０に戻すことが可能である（アンインストールともいう）。

【００６７】

この場合、コントローラ５０のＩＣカード用ドライバ５２では、アンインストールするソフトウェアがＩＣカード１０上のソフトウェアとそのソフトウェアのシリアルナンバーが同一であることを認識し、コントローラ５０上のソフトウェアを削除して、ＩＣカード１０にそのソフトウェアのインストールカウンタを１増やす（インクリメントする）命令を送信する。この命令を受けてＩＣカード１０上のＣＰＵ１２は、インストールカウンタがインクリメントされる。

【００６８】

このようにしたため、ＩＣカード１０からコントローラ５０にソフトウェアをインストールしても、そのインストールされたソフトウェアをコントローラ５０からアンインストールした場合は、ＩＣカード１０のメモリ１４のインストールカウンタのカウント値がインクリメントされるため、再び別のコントローラにインストールすることが可能となる。従って、ソフトウェアの配布本数制限をインストールカウンタにより厳密に管理することができる。

【００６９】

さらに、この実施の形態４では、図１に示されるＩＣカード１０のメモリ１４のソフトウェア領域内には、ソフトウェアの実行を許可するか否かを表す実行許可フラグが格納される実行フラグ領域３０が設けられている。このため、図３に示されるように、コントローラ５０側のＩＣカード用ドライバ５２は、ＩＣカード１０のメモリ１４上のソフトウエアの実行要求に対して実行許可フラグを確認し、ソフトウェアの実行が許可されている場合はその実行命令をＩＣカード１０に転送するが、ソフトウェアの実行が許可されない場合は実行要求を発行しないようにする。

【００７０】

このＩＣカード１０の実行フラグ領域３０内の実行許可フラグを変更する場合は、ここではパスワードを必要とする。すなわち、実行許可フラグを変更する際に、ＩＣカード用ドライバ５２において不図示の入力部から入力されるパスワードの正誤が判断され、パスワードが正しい場合にのみ実行許可フラグの許可と不許可の変更を行うことができ、その変更結果をＩＣカード１０上のメモリ１４に書き込まれる。

【００７１】

このようにしたため、ソフトウェアの誤使用を防ぐことが可能となり、さらにその実行許可フラグの設定はパスワードによって行われるため、実行許可フラグの設定を厳密に管理することができる。

【００７２】

また、図７には、複数のソフトウェアが格納可能なメモリ７０を備えたＩＣカード１０

の概略構成が示されている。このメモリ７０の特徴は、複数（ｎ個）のソフトウェアを保存することが可能なソフトウェア１領域７２ａ～ソフトウェアｎ領域７２ｃが確保されていることにある。

【００７３】

そして、上述したように、ソフトウェアにインストール可能回数を規定するインストールカウンタやソフトウェアの実行の可否を表す実行許可フラグの設定、あるいは、書き込み禁止の有無を設定するライトプロテクトフラグ等の設定を複数の各ソフトウェアに対してそれぞれ行うことができる。

【００７４】

このようにしたため、ＩＣカード１０のメモリ７０に複数のソフトウェアを格納しておいて配布することが可能であり、それらの各ソフトウェア毎にインストールカウンタの設定や実行許可フラグ等の設定を行うことが可能であり、例えば、インストール禁止のサーバーソフトと無制限にインストール可能なクランアントソフトというような配布条件が異なるソフトウェアを一枚のＩＣカードで提供することが可能となる。

【００７５】

また、図３に示されるコントローラ５０からＩＣカード１０に対して、コントローラ上のソフトウェアを転送して格納することもできる。

【００７６】

その場合には、まず、図３に示されるＩＣカード１０をコントローラ５０に装着すると、コントローラ５０は、ＩＣカード用ドライバ５２においてＩＣカードの装着を検出し、ＩＣカード１０との間で通信を開始することができる。ＩＣカード用ドライバ５２は、ユーザからのソフトウェアの格納要求に対してインストールカウンタの設定、実行許可フラグの設定およびそれらを変更する際のパスワードの入力を要求する。そして、それらが設定されるとその情報とともにＩＣカード１０上のメモリ１４にソフトウェアを格納することができる。

【００７７】

このように、ＩＣカード１０のメモリ１４に新たにソフトウェアを書き込む場合は、インストールカウンタと実行フラグとをパスワードを使って設定を行うようにしたため、ソフトウェアを安全かつ厳密に管理することが可能となる。

【００７８】

なお、上記した各実施の形態において、ソフトウェアが格納されたＩＣカードを装着して配布する対象をコントローラとして説明したが、必ずしもこれに限定されるものではなく、パーソナルコンピュータ等に装着してパーソナルコンピュータに対してソフトウェアを配布するものであっても良い。

【００８０】

【発明の効果】

以上説明したように、この発明に係るＩＣカードシステムによれば、ソフトウェアは、必ずしもコントローラ側の制御部に依存しないソフトウェアで作成することができるので、自由度が大きく、さらにＩＣカードとコントローラとの間で資源情報が一致していれば、ＩＣカードのソフトウェアをコントローラ側で実行することもできる。

【００８１】

つぎの発明に係るＩＣカードシステムによれば、ＩＣカード制御部でコントローラ上のソフトウェアを実行することができるので、コントローラとＩＣカード内で異なるソフトウェアをそれぞれ実行させて並列処理することも可能となり、処理速度を向上させることができる。

【００８２】

つぎの発明に係るＩＣカードシステムによれば、ソフトウェアの配布前にインストール可能回数をインストールカウンタに設定して、一旦コントローラ側にインストールすればＩＣカードを装着しなくてもソフトウェアを実行することが可能であり、インストールカウンタに設定された数以上のインストールを禁止することができるので、ソフトウェアの

無制限なコピーを防止することができ、厳密なソフトウェアの配布本数制限が可能となる。さらに、インストール可能回数の設定は、パスワードを用いて行うので、ソフトウェアの配布本数制限をさらに厳密にすることができる。

【００８３】

つぎの発明に係るＩＣカードシステムによれば、ソフトウェアの配布本数制限をインストールカウンタにより厳密に管理することができる。

【００８４】

つぎの発明に係るＩＣカードシステムによれば、ＩＣカードメモリに複数のソフトウェアが格納されて配布可能であり、それらの各ソフトウェア毎にインストールカウンタの設定や実行許可フラグを設定することができるので、様々な属性を持ったソフトウェアを一枚のＩＣカードで同時に配布することが可能となる。

【図面の簡単な説明】

【図１】　実施の形態１に係るＩＣカードシステムのＩＣカードの概略構成図である。

【図２】　図１のアーキテクチャ情報領域に格納されている情報例である。

【図３】　図１のＩＣカードとこれが装着されるコントローラとで構成されるＩＣカードシステムの概略構成図である。

【図４】　実施の形態１に係るソフトウェアの実行動作を説明するフローチャートである。

【図５】　実施の形態２に係るソフトウェアの実行動作を説明するフローチャートである。

【図６】　実施の形態３に係るソフトウェアの実行動作を説明するフローチャートである。

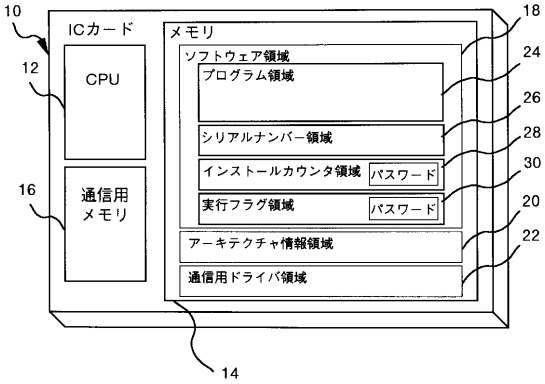
【図７】　実施の形態４に係るＩＣカードシステムのＩＣカードの概略構成図である。

【図８】　従来のソフトウェアが格納されたフロッピー（Ｒ）ディスクとコントローラとの概略構成図である。

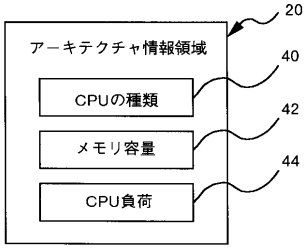
【符号の説明】

１０　ＩＣカード，１２　ＣＰＵ（ＩＣカード制御部），１４　メモリ（ＩＣカードメモリ），１６　通信用メモリ，２２　通信用ドライバ領域（通信用ドライバ），５２　ＩＣカード用ドライバ（通信用ドライバ），５４　メモリ（コントローラメモリ），５６　ＣＰＵ（コントローラ制御部）。

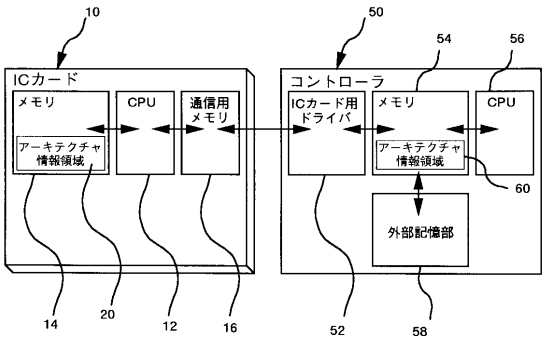
【図１】


10
ICカード
12
CPU
16
通信用
メモリ
メモリ
ソフトウェア領域
プログラム領域
シリアルナンバー領域
インストールカウンタ領域
パスワード
実行フラグ領域
パスワード
アーキテクチャ情報領域
通信用ドライバ領域
14
18
24
26
28
30
20
22

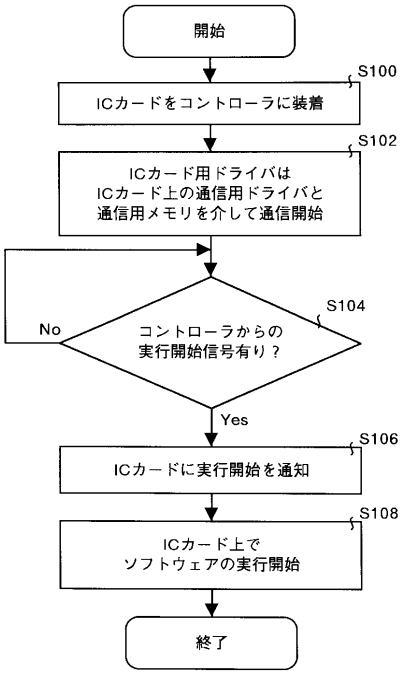
【図２】


20
アーキテクチャ情報領域
40
CPUの種類
42
メモリ容量
44
CPU負荷

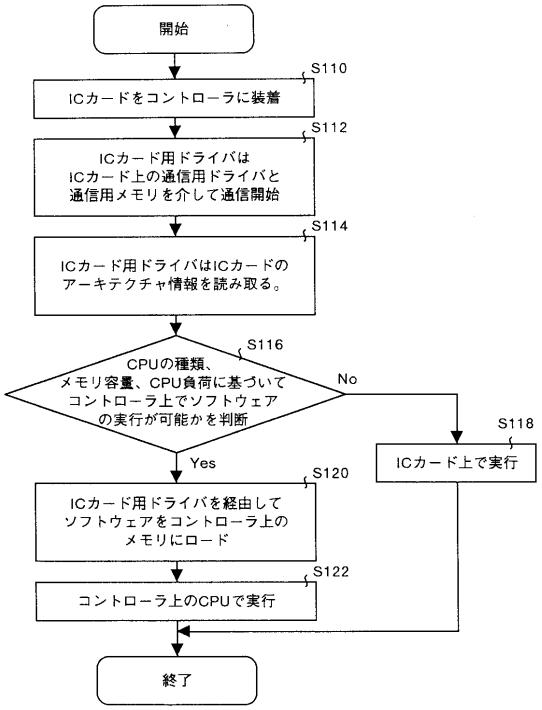
【図３】


10
ICカード
メモリ
アーキテクチャ
情報領域
CPU
通信用
メモリ
50
コントローラ
ICカード用
ドライバ
54
メモリ
アーキテクチャ
情報領域
56
CPU
60
外部記憶部
14
20
12
16
52
58

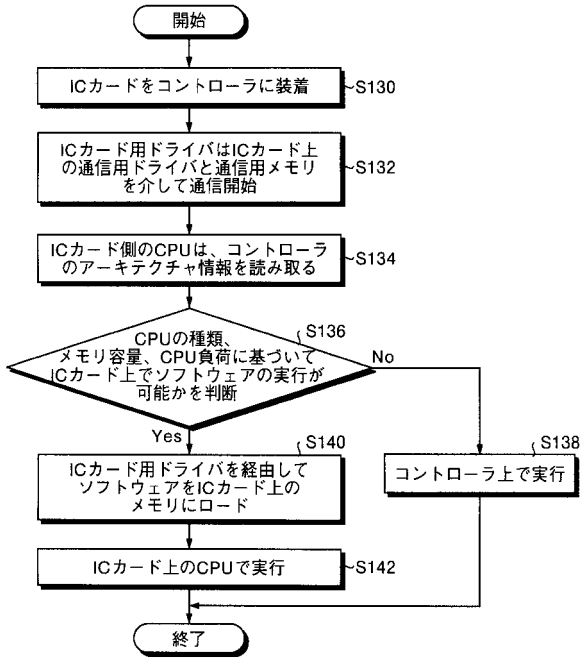
【図４】


開始
S100
ICカードをコントローラに装着
S102
ICカード用ドライバは
ICカード上の通信用ドライバと
通信用メモリを介して通信開始
S104
No
コントローラからの
実行開始信号有り？
Yes
S106
ICカードに実行開始を通知
S108
ICカード上で
ソフトウェアの実行開始
終了

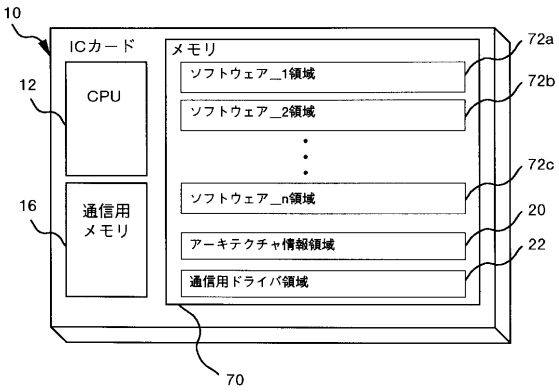
【図５】


開始
S110
ICカードをコントローラに装着
S112
ICカード用ドライバは
ICカード上の通信用ドライバと
通信用メモリを介して通信開始
S114
ICカード用ドライバはICカードの
アーキテクチャ情報を読み取る。
S116
CPUの種類、
メモリ容量、CPU負荷に基づいて
コントローラ上でソフトウェア
の実行が可能かを判断
No
S118
ICカード上で実行
Yes
S120
ICカード用ドライバを経由して
ソフトウェアをコントローラ上の
メモリにロード
S122
コントローラ上のCPUで実行
終了

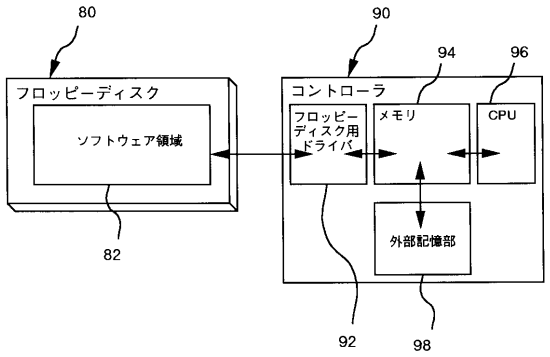
【図６】


開始
ICカードをコントローラに装着
S130
ICカード用ドライバはICカード上の通信用ドライバと通信用メモリを介して通信開始
S132
ICカード側のCPUは、コントローラのアーキテクチャ情報を読み取る
S134
CPUの種類、メモリ容量、CPU負荷に基づいてICカード上でソフトウェアの実行が可能かを判断
S136
No
Yes
S140
ICカード用ドライバを経由してソフトウェアをICカード上のメモリにロード
S138
コントローラ上で実行
ICカード上のCPUで実行
S142
終了

【図７】


10
ICカード
メモリ
12
CPU
16
通信用メモリ
ソフトウェア_1領域
ソフトウェア_2領域
ソフトウェア_n領域
アーキテクチャ情報領域
通信用ドライバ領域
72a
72b
72c
20
22
70

【図８】


80
フロッピーディスク
ソフトウェア領域
82
90
94
96
コントローラ
フロッピーディスク用ドライバ
メモリ
CPU
外部記憶部
92
98

フロントページの続き

(56)参考文献 特開平０４－０３７９７５（ＪＰ，Ａ）
特開平０７－０４４６７２（ＪＰ，Ａ）
国際公開第９６／０３８８２６（WO，Ａ１）
特開平０１－０９３８７７（ＪＰ，Ａ）
特開昭６１－２５１９８５（ＪＰ，Ａ）
特開平０５－０２３３１２（ＪＰ，Ａ）
国際公開第９７／００７４４８（WO，Ａ１）
特表平１１－５０５９４７（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.[7]，ＤＢ名)
G06K 17/00
G06F 9/06 550
G06F 12/14 320
G06K 19/07