

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4213258号
(P4213258)

(45) 発行日 平成21年1月21日(2009.1.21)

(24) 登録日 平成20年11月7日(2008.11.7)

(51) Int.Cl.

F I

G O 6 K 19/07 (2006.01)

G O 6 K 19/00 N

G O 6 K 17/00 (2006.01)

G O 6 K 17/00 B

G O 6 K 17/00 D

請求項の数 13 (全 30 頁)

(21) 出願番号	特願平10-201805	(73) 特許権者	000003078
(22) 出願日	平成10年7月16日(1998.7.16)		株式会社東芝
(65) 公開番号	特開2000-36022(P2000-36022A)		東京都港区芝浦一丁目1番1号
(43) 公開日	平成12年2月2日(2000.2.2)	(74) 代理人	100058479
審査請求日	平成17年7月15日(2005.7.15)		弁理士 鈴江 武彦
		(74) 代理人	100084618
			弁理士 村松 貞男
		(74) 代理人	100092196
			弁理士 橋本 良郎
		(74) 代理人	100091351
			弁理士 河野 哲
		(74) 代理人	100088683
			弁理士 中村 誠
		(74) 代理人	100070437
			弁理士 河井 将次

最終頁に続く

(54) 【発明の名称】 ICカード、ICカード処理システム、及びICカード処理方法

(57) 【特許請求の範囲】

【請求項1】

ICチップを有するICカードにおいて、

外部から送信されるデータを受信する通信手段と、

メインファイル、このメインファイルの配下に位置づけられたサブファイル、前記メインファイルの定義情報、及び前記サブファイルの定義情報を格納する格納手段と、

前記通信手段により受信されたデータが、所定のファイルの削除を命じるコマンドである場合、前記格納手段の中から、前記所定のファイル、及びこの所定のファイルの定義情報を削除する削除手段と、

を備え、

前記削除手段は、前記通信手段により受信されたデータが、所定のメインファイルの削除を命じるコマンドである場合、前記格納手段の中から、前記所定のメインファイル、この所定のメインファイルの配下に位置づけられた所定のサブファイル、及びこれら所定のメインファイルの定義情報と所定のサブファイルの定義情報とを削除し、

さらに、

前記格納手段に対して前記定義情報を格納する際に、前記格納手段が有する格納領域の一端から順に前記定義情報を配置し、前記格納手段に対して前記ファイルを格納する際に、前記格納領域の他端から順に前記ファイルを配置する第1の配置手段と、

前記削除手段による前記定義情報の削除に伴い、前記格納領域の一端から順に前記定義情報を再配置し、前記削除手段による前記ファイルの削除に伴い、前記格納領域の他端か

10

20

ら順に前記ファイルを再配置する第2の配置手段と、

最大n個のメインファイルを管理するためのnビットのビットマップデータを格納し、前記格納手段に格納された各メインファイルに対して前記ビットマップデータのうちの1ビットを割り当てて割り当てられた1ビットを1に設定し各メインファイルを管理し、前記所定のメインファイルの削除に対応してこの所定のメインファイルの管理に割り当てられていた1ビットを0に設定するメインファイル管理手段と、

を備えたことを特徴とするICカード。

【請求項2】

前記第1の配置手段により前記定義情報が第1の位置に配置されたのに基づきこの第1の位置を示す第1のアドレスデータを管理し、前記第2の配置手段により前記定義情報が第1の位置から第2の位置に再配置されたのに基づき前記第1のアドレスデータを前記第2の位置を示す第2のアドレスデータに更新して管理し、前記第1の配置手段により前記ファイルが第3の位置に配置されたのに基づきこの第3の位置を示す第3のアドレスデータを管理し、前記第2の配置手段により前記ファイルが第3の位置から第4の位置に再配置されたのに基づき前記第3のアドレスデータを前記第4の位置を示す第4のアドレスデータに更新して管理するアドレス管理手段を備えたことを特徴とする請求項1に記載のICカード。

10

【請求項3】

前記通信手段により受信されたデータが、第1及び第2のファイルの削除を命じるコマンドである場合、前記削除手段により前記格納手段の中から前記第1のファイルを削除させ、前記第2の配置手段により再配置を実行させてから、前記削除手段により前記第1のファイルの定義情報を削除させ、前記第2の配置手段により再配置を実行させてから、前記削除手段により前記第2のファイルを削除させ、前記第2の配置手段により再配置を実行させてから、前記削除手段により前記第2のファイルの定義情報を削除させ、前記第2の配置手段により再配置を実行させて、前記コマンドの命令を完了する削除制御手段を備えたことを特徴とする請求項1に記載のICカード。

20

【請求項4】

前記通信手段により受信されたデータが、第1及び第2のファイルの削除を命じるコマンドである場合、前記削除手段により前記格納手段の中から前記第1のファイルを削除させ、前記第2の配置手段により再配置を実行させてから、前記削除手段により前記第2のファイルを削除させ、前記第2の配置手段により再配置を実行させてから、前記削除手段により前記第1のファイルの定義情報を削除させ、前記第2の配置手段により再配置を実行させてから、前記削除手段により前記第2のファイルの定義情報を削除させ、前記第2の配置手段により再配置を実行させて、前記コマンドの命令を完了する削除制御手段を備えたことを特徴とする請求項1に記載のICカード。

30

【請求項5】

前記削除手段、並びに前記第1及び第2の配置手段による処理の実行記録を格納する実行記録格納手段と、

前記実行記録格納手段に格納された実行記録をバックアップする実行記録バックアップ手段と、

40

を備えたことを特徴とする請求項1に記載のICカード。

【請求項6】

第1のファイル、及びこの第1のファイルのアクセスに必要なキーデータを保持した第2のファイルを格納するファイル格納手段と、

前記第1のファイルのアクセス要求に伴い入力されたキーデータと前記第2のファイルに格納されたキーデータとを照合して、これら両キーデータが一致したのに基づき前記第1のファイルのアクセスを許可し、前記削除手段により前記第2のファイルが削除されたのに基づき、前記第1のファイルのアクセスを不能とするファイルアクセス管理手段と、

を備えたことを特徴とする請求項1に記載のICカード。

【請求項7】

50

請求項 1 から 6 の一つに記載の IC カード、及びこの IC カードを処理する IC カード処理装置を備えた IC カード処理システムであって、

前記 IC カード処理装置は、

削除を命じる削除コマンドを生成するコマンド生成手段と、

前記コマンド生成手段により生成された前記削除コマンドを送信する第 2 の通信手段と

、

を備え、

前記 IC カードの前記通信手段は、前記第 2 の通信手段により送信された前記削除コマンドを受信する、ことを特徴とする IC カード処理システム。

【請求項 8】

IC チップを有する IC カードに適用される IC カード処理方法は、

外部から送信されるデータを受信する第 1 のステップと、

前記第 1 のステップにより受信されたデータが、所定のファイルの削除を命じるコマンドである場合、データ格納部の中から、前記所定のファイル、及びこの所定のファイルの定義情報を削除する第 2 のステップと、

を備え、

前記第 2 のステップは、

前記第 1 のステップにより受信されたデータが、所定のメインファイルの削除を命じるコマンドである場合、データ格納部の中から、前記所定のメインファイル、この所定のメインファイルの配下に位置づけされた所定のサブファイル、及びこれら所定のメインファイルの定義情報と所定のサブファイルの定義情報とを削除するステップを備え、

さらに、IC カード処理方法は、

前記データ格納部に対して前記定義情報を格納するときに、前記データ格納部が有する格納領域の一端から順に前記定義情報を配置し、前記データ格納部に対して前記ファイルを格納するときに、前記格納領域の他端から順に前記ファイルを配置する第 3 のステップと、

前記第 2 のステップによる前記定義情報の削除に伴い、前記格納領域の一端から順に前記定義情報を再配置し、前記第 2 のステップによる前記ファイルの削除に伴い、前記格納領域の他端から順に前記ファイルを再配置する第 4 のステップと、

を備え、

さらに、IC カード処理方法は、

最大 n 個のメインファイルを管理するための n ビットのビットマップデータを格納し、前記格納された各メインファイルに対して前記ビットマップデータのうちの 1 ビットを割り当てて割り当てられた 1 ビットを 1 に設定し各メインファイルを管理し、前記所定のメインファイルの削除に対応してこの所定のメインファイルの管理に割り当てられていた 1 ビットを 0 に設定することを特徴とする IC カード処理方法。

【請求項 9】

前記第 3 のステップにより前記定義情報が第 1 の位置に配置されたのに基づきこの第 1 の位置を示す第 1 のアドレスデータを管理し、前記第 4 のステップにより前記定義情報が第 1 の位置から第 2 の位置に再配置されたのに基づき前記第 1 のアドレスデータを前記第 2 の位置を示す第 2 のアドレスデータに更新して管理し、前記第 3 のステップにより前記ファイルが第 3 の位置に配置されたのに基づきこの第 3 の位置を示す第 3 のアドレスデータを管理し、前記第 4 のステップにより前記ファイルが第 3 の位置から第 4 の位置に再配置されたのに基づき前記第 3 のアドレスデータを前記第 4 の位置を示す第 4 のアドレスデータに更新して管理する第 5 のステップを備えたことを特徴とする請求項 8 に記載の IC カード処理方法。

【請求項 10】

前記第 1 のステップにより受信されたデータが、第 1 及び第 2 のファイルの削除を命じるコマンドである場合、前記第 2 のステップにより前記データ格納部の中から前記第 1 のファイルを削除させ、前記第 4 のステップにより再配置を実行させてから、前記第 2 のス

10

20

30

40

50

テップにより前記第 1 のファイルの定義情報を削除させ、前記第 4 のステップにより再配置を実行させてから、前記第 2 のステップにより前記第 2 のファイルを削除させ、前記第 4 のステップにより再配置を実行させてから、前記第 2 のステップにより前記第 2 のファイルの定義情報を削除させ、前記第 4 のステップにより再配置を実行させて、前記コマンドの命令を完了する第 5 のステップを備えたことを特徴とする請求項 8 に記載の IC カード処理方法。

【請求項 11】

前記第 1 のステップにより受信されたデータが、第 1 及び第 2 のファイルの削除を命じるコマンドである場合、前記第 2 のステップにより前記データ格納部の中から前記第 1 のファイルを削除させ、前記第 4 のステップにより再配置を実行させてから、前記第 2 のステップにより前記第 2 のファイルを削除させ、前記第 4 のステップにより再配置を実行させてから、前記第 2 のステップにより前記第 1 のファイルの定義情報を削除させ、前記第 4 のステップにより再配置を実行させてから、前記第 2 のステップにより前記第 2 のファイルの定義情報を削除させ、前記第 4 のステップにより再配置を実行させて、前記コマンドの命令を完了する第 5 のステップを備えたことを特徴とする請求項 8 に記載の IC カード処理方法。

10

【請求項 12】

前記第 2 のステップ、並びに前記第 3 及び第 4 のステップによる処理の実行記録を格納する第 5 のステップと、

前記第 5 のステップにより格納された実行記録をバックアップする第 6 のステップと、
を備えたことを特徴とする請求項 8 に記載の IC カード処理方法。

20

【請求項 13】

第 1 のファイルのアクセス要求に伴い入力されたキーデータと、第 2 のファイルに格納された前記第 1 のファイルのアクセスに必要なキーデータとを照合して、これら両キーデータが一致したのに基づき前記第 1 のファイルのアクセスを許可し、前記第 2 のステップにより前記第 2 のファイルが削除されたのに基づき、前記第 1 のファイルのアクセスを不能とするステップを備えたことを特徴とする請求項 8 に記載の IC カード処理方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

30

この発明は、カードリーダー・ライタから送信されるコマンドを受信して、この受信したコマンドに従った処理を実行する IC カードに関する。

【0002】

また、この発明は、IC カードに対してコマンドを送信するカードリーダー・ライタ、及びこのカードリーダー・ライタから送信されるコマンドを受信して、この受信したコマンドに従った処理を実行する IC カードを備えた IC カード処理システムに関する。

【0003】

さらに、この発明は、カードリーダー・ライタから送信されるコマンドを受信して、この受信したコマンドに従った処理を実行する IC カードに適用される IC カード処理方法に関する。

40

【0004】

【従来の技術】

近年、携帯可能な情報記憶媒体として IC カードが注目されている。この IC カードには、不揮発性メモリ及び制御部を備えた IC チップが内蔵されており、不揮発性メモリには様々なアプリケーションを実行するためのファイルが記憶されている。

【0005】

IC カードは、カードリーダー・ライタから送信されるコマンドを受けて、このコマンドに従い所定のアプリケーションを実行する。つまり、不揮発性メモリに記憶されている所定のファイルを読み出し、所定のアプリケーションを実行する。

【0006】

50

また、カードリーダー・ライターにより、ＩＣカードの不揮発性メモリに対して、新たなファイルを記憶させることにより、ＩＣカードに新たなアプリケーションを追加することができる。

【０００７】

【発明が解決しようとする課題】

当然ながら、ＩＣカードの不揮発性メモリの空き容量には限度があるため、ＩＣカードに追加できるアプリケーションにも限度が生じる。このことが、ＩＣカードの汎用性を阻害する要因の一つになっていた。

【０００８】

この発明の目的は、上記したような事情に鑑み成されたものであって、新たなアプリケーションの設定の自由度に優れたＩＣカード、ＩＣカード処理システム、ＩＣカード処理方法を提供することにある。

10

【０００９】

【課題を解決するための手段】

上記課題を解決し目的を達成するために、この発明のＩＣカード、ＩＣカード処理システム、及びＩＣカード処理方法は、以下のように構成されている。

【００１０】

この発明は、ＩＣチップを有するＩＣカードにおいて、外部から送信されるデータを受信する通信手段と、データを格納する格納手段と、前記通信手段により受信されたデータが、所定のデータの削除を命じるコマンドである場合、前記格納手段に格納されたデータの中から前記所定のデータを削除する削除手段とを備えている。

20

【００１１】

この発明は、ＩＣチップを有するＩＣカードにおいて、外部から送信されるデータを受信する通信手段と、ファイル及びこのファイルの定義情報を格納する格納手段と、前記通信手段により受信されたデータが、所定のファイルの削除を命じるコマンドである場合、前記格納手段の中から、前記所定のファイル、及びこの所定のファイルの定義情報を削除する削除手段とを備えている。

【００１２】

この発明は、ＩＣチップを有するＩＣカードにおいて、外部から送信されるデータを受信する通信手段と、メインファイル、このメインファイルの配下に位置づけされたサブファイル、及び各ファイルの定義情報を格納する格納手段と、前記通信手段により受信されたデータが、所定のメインファイルの削除を命じるコマンドである場合、前記格納手段の中から、前記所定のメインファイル、この所定のメインファイルの配下に位置づけされた所定のサブファイル、及びこれら所定のメインファイルの定義情報と所定のサブファイルの定義情報とを削除する削除手段とを備えている。

30

【００１３】

この発明は、ＩＣカード、及びこのＩＣカードを処理するＩＣカード処理装置を備えたＩＣカード処理システムにおいて、所定のファイルの選択を命じる選択コマンド、及び削除を命じる削除コマンドを生成するコマンド生成手段と、前記コマンド生成手段により生成された前記選択コマンド及び前記削除コマンドを送信する第１の通信手段と、前記第１の通信手段により送信された前記選択コマンド及び前記削除コマンドを受信する第２の通信手段と、ファイル及びこのファイルの定義情報を格納する格納手段と、前記第２の通信手段により受信された前記選択コマンドの命令に従い、前記格納手段の中から前記所定のファイルを選択し、前記第２の通信手段により受信された前記削除コマンドに従い、この選択された前記所定のファイル、及びこの所定のファイルの定義情報を前記格納手段から削除する削除手段とを備えている。

40

【００１４】

この発明は、ＩＣカード、及びこのＩＣカードを処理するＩＣカード処理装置を備えたＩＣカード処理システムにおいて、所定のファイルの削除を命じる削除コマンドを生成するコマンド生成手段と、前記コマンド生成手段により生成された前記削除コマンドを送信す

50

る第１の通信手段と、前記第１の通信手段により送信された前記削除コマンドを受信する第２の通信手段と、ファイル及びこのファイルの定義情報を格納する格納手段と、前記第２の通信手段により受信された前記削除コマンドの命令に従い、前記格納手段に格納されている前記所定のファイル、及びこの所定のファイルの定義情報を削除する削除手段とを備えている。

【００１５】

この発明は、ＩＣカード、及びこのＩＣカードを処理するＩＣカード処理装置を備えたＩＣカード処理システムにおいて、所定のメインファイルの選択を命じる選択コマンド、及び削除を命じる削除コマンドを生成するコマンド生成手段と、前記コマンド生成手段により生成された前記選択コマンド及び前記削除コマンドを送信する第１の通信手段と、前記第１の通信手段により送信された前記選択コマンド及び前記削除コマンドを受信する第２の通信手段と、メインファイル、このメインファイルの配下に位置づけされたサブファイル、及び各ファイルの定義情報を格納する格納手段と、前記第２の通信手段により受信された前記選択コマンドの命令に従い、前記格納手段の中から前記所定のメインファイルを選択し、前記第２の通信手段により受信された前記削除コマンドに従い、この選択された前記所定のメインファイル、この所定のメインファイルの配下に位置づけされた所定のサブファイル、及びこれら所定のメインファイルの定義情報と所定のサブファイルの定義情報とを削除する削除手段とを備えている。

【００１６】

この発明は、ＩＣカード、及びこのＩＣカードを処理するＩＣカード処理装置を備えたＩＣカード処理システムにおいて、所定のメインファイルの削除を命じる削除コマンドを生成するコマンド生成手段と、前記コマンド生成手段により生成された前記削除コマンドを送信する第１の通信手段と、前記第１の通信手段により送信された前記削除コマンドを受信する第２の通信手段と、メインファイル、このメインファイルの配下に位置づけされたサブファイル、及び各ファイルの定義情報を格納する格納手段と、前記第２の通信手段により受信された前記削除コマンドの命令に従い、前記格納手段に格納されている前記所定のメインファイル、この所定のメインファイルの配下に位置づけされた所定のサブファイル、及びこれら所定のメインファイルの定義情報と所定のサブファイルの定義情報とを削除する削除手段とを備えている。

【００１７】

この発明は、ＩＣチップを有するＩＣカードに適用される方法であって、外部から送信されるデータを受信する第１のステップと、前記第１のステップにより受信されたデータが、所定のデータの削除を命じるコマンドである場合、データ格納部に格納されたデータの中から前記所定のデータを削除する第２のステップとを備えている。

【００１８】

この発明は、ＩＣチップを有するＩＣカードに適用される方法であって、外部から送信されるデータを受信する第１のステップと、前記第１のステップにより受信されたデータが、所定のファイルの削除を命じるコマンドである場合、データ格納部の中から、前記所定のファイル、及びこの所定のファイルの定義情報を削除する第２のステップとを備えている。

【００１９】

この発明は、ＩＣチップを有するＩＣカードに適用される方法であって、外部から送信されるデータを受信する第１のステップと、前記第１のステップにより受信されたデータが、所定のメインファイルの削除を命じるコマンドである場合、データ格納部の中から、前記所定のメインファイル、この所定のメインファイルの配下に位置づけされた所定のサブファイル、及びこれら所定のメインファイルの定義情報と所定のサブファイルの定義情報とを削除する第２のステップとを備えている。

【００２０】

【発明の実施の形態】

以下、この発明の実施の形態について図面を参照して説明する。

【 0 0 2 1 】

図 1 は、この発明の I C カード処理システムの概略構成を示すブロック図である。図 1 に示すように、I C カード処理システム 1 は、I C カード 1 0 0 及び I C カード処理装置 2 0 0 を備えている。

【 0 0 2 2 】

I C カード 1 0 0 には、制御部 1 1 0、メモリ 1 2 0、及びコンタクト部 1 3 0 が設けられている。このうち、制御部 1 1 0 及びメモリ 1 2 0 は、I C チップとしてカードに埋め込まれている。

【 0 0 2 3 】

制御部 1 1 0 は、例えば C P U で構成されており、メモリ 1 2 0 のプログラムメモリ 1 2 2 に記憶されているプログラムに基づいて、所定の処理を実行する。この処理の処理の中に、後述するディレクトリ削除処理が含まれる。つまり、この制御部 1 1 0 は、削除手段として機能する。

10

【 0 0 2 4 】

メモリ 1 2 0 には、プログラムメモリ 1 2 2、データメモリ 1 2 4、及びワーキングメモリ 1 2 6 が含まれている。

【 0 0 2 5 】

プログラムメモリ 1 2 2 は、例えばマスク R O M で構成されており、このプログラムメモリ 1 2 2 には、制御部 1 1 0 による所定の処理の実行に必要なとされるプログラムなどが記録されている。つまり、このプログラムメモリ 1 2 2 には、ディレクトリ削除処理に必要なとされるプログラムも記録されている。

20

【 0 0 2 6 】

データメモリ 1 2 4 (格納手段) は、例えば E E P R O M で構成されており、このデータメモリ 1 2 4 には、各種アプリケーションを実現させるためのアプリケーションファイルなどが記憶されている。詳しく言うと、データメモリ 1 2 4 には、図 2 に示すようなファイル構造を構築するための定義情報が記憶されており、このファイル構造の下でアプリケーションファイルが管理されている。ここで、図 2 に示すファイル構造を簡単に説明すると、M F (main file) の配下には、D F (dedicated file) 1 及び D F 2 が位置づけされている。D F 1 の配下には、D F 1 - 1 及び E F (elementally file) 1 - 1 が位置づけされており、さらに D F 1 - 1 の配下には、E F 1 - 1 - 1 が位置づけされている。D F 2 の配下には、E F 2 - 1 及び E F 2 - 2 が位置づけされている。例えば、E F 1 - 1 及び E F 1 - 1 - 1 には、第 1 のアプリケーションを実現させるためのアプリケーションファイル (e f 1 - 1 及び e f 1 - 1 - 1) が含まれている。また、E F 2 - 1 及び E F 2 - 2 には、第 1 のアプリケーションと異なる第 2 のアプリケーションを実現させるためのアプリケーションファイル (e f 2 - 1 及び e f 2 - 2) が含まれている。因みに、D F と E F は、メインファイルとサブファイルの関係にあると言える。

30

【 0 0 2 7 】

ワーキングメモリ 1 2 6 は、例えば R A M で構成されており、このワーキングメモリ 1 2 6 には、制御部 1 1 0 の処理データなどが一時的に記憶される。

【 0 0 2 8 】

コンタクト部 1 3 0 (通信手段) は、カード処理装置 2 0 0 のリーダ・ライタ 2 3 0 との電氣的接触により、I C カード 1 0 0 に記憶されたデータをリーダ・ライタ 2 3 0 に提供したり、リーダ・ライタ 2 3 0 から提供されるデータを受け取ったりする。

40

【 0 0 2 9 】

I C カード処理装置 2 0 0 は、金融システム又はショッピングシステムなどの端末装置として適用されるものである。また、この I C カード処理装置 2 0 0 には、店頭に設置して運用される設置型のタイプと、携帯可能な携帯型のタイプとがある。I C カード処理装置 2 0 0 には、制御部 2 1 0、メモリ 2 2 0、リーダ・ライタ 2 3 0、入力部 2 4 0、及びディスプレイ 2 5 0 が設けられている。

【 0 0 3 0 】

50

制御部 210 は、例えば CPU で構成されており、メモリ 220 に記憶されているプログラムに基づき、所定の処理を実行する。この所定の処理には、例えば、後述するセレクトコマンド及びディレクトリ削除コマンドを生成するコマンド生成処理などがある。つまり、この制御部 210 は、コマンドを生成するコマンド生成手段として機能する。

【0031】

メモリ 220 には、制御部 210 による所定の処理の実行に必要とされるプログラムが記憶されている。例えば、コマンド生成処理の実行に必要とされるプログラムが記憶されている。

【0032】

リーダ・ライタ 230（通信手段）は、ICカード 100 のコンタクト部 130 を介して、ICカード 100 に記憶されているデータを読み出したり、ICカード 100 に対してデータを書き込んだりする。

10

【0033】

入力部 240 は、ICカード処理装置の操作者による入力を受け付ける。例えば、後述するバックアップ領域のサイズの設定（変更）を受け付ける。この入力部 240 で受け付けられたバックアップ領域のサイズの設定（変更）に基づき、リーダ・ライタ 230 を介して ICカード 100 に対して、バックアップ領域のサイズの設定（変更）が指示される。ICカード 100 側では、この指示に基づきバックアップ領域のサイズが決定される。つまり、入力部 240 は、バックアップ領域のサイズを変更するサイズ変更手段として機能する。

20

【0034】

ディスプレイ 250 は、例えば、ICカード 100 から読み出されたデータを表示する。

【0035】

次に、上記説明した ICカードのデータメモリ 124 に記憶されたアプリケーションファイルの削除について説明する。

【0036】

この発明は、所定のディレクトリ（図 2 に示す DF）を削除することにより、この所定のディレクトリの配下に位置づけられたサブディレクトリ（EF）の削除を実現するというものである。アプリケーションの削除には二つの方法があり、一つはカレント DF の削除であり、もう一つは指定 DF の削除である。

30

【0037】

どちらの削除方法を採用するにしても、ICカード 100 側において、ICカード処理装置 200 から出力されるコマンドが受信できなければならない。つまり、ICカード 100 が、ICカード処理装置 200 にセットされなければならない。ICカード処理装置 200 に ICカード 100 がセットされると、ICカード処理装置 200 により ICカード 100 のデータメモリ 124 内のファイルにアクセスすることができる。

【0038】

例えば、入力部 240 を介して DF 2 のファイル選択が指示されると、制御部 210 の制御によりセレクトコマンドが生成され、リーダ・ライタ 230 からコンタクト部 130 に対して、生成されたセレクトコマンドとファイル名（DF 2）が送信される。コンタクト部 130 でセレクトコマンドとファイル名（DF 2）が受信されると、制御部 110 の制御により、セレクトコマンドと共に送られてきたファイル名（DF 2）のファイルがデータメモリ 124 から選択され、この選択されたファイルがコンタクト部 130 を介してリーダ・ライタ 230 に送信される。また、制御部 110 は、選択中のファイル（DF 2）をカレントファイルとして認識する。制御部 210 は、先に送信したセレクトコマンドのレスポンスとして、コンタクト部 130 から送信されるファイルをリーダ・ライタ 230 を介して受信し、この受信されたファイルをディスプレイ 250 に出力する。

40

【0039】

ここで、入力部 240 を介してカレント DF の削除が指示されると、制御部 210 の制御によりディレクトリ削除コマンドが生成され、リーダ・ライタ 230 からコンタクト部 1

50

30に対して、生成されたディレクトリ削除コマンドのみが送信される。コンタクト部130でディレクトリ削除コマンドのみが受信されると、制御部110の制御によりデータメモリ124に記憶されているカレントファイル（この場合はDF2）が削除される。つまり、カレントファイル（この場合はDF2）が削除されることにより、このカレントファイルの配下に位置づけられたサブファイル（この場合はEF2-1及びEF2-2）が削除されることになる。カレントファイル（この場合はDF2）の削除が完了すると、削除されたカレントファイルの上位に位置づけされたファイル（この場合はMF）がカレントファイルとして認識される。

【0040】

続いて、指定DF（例えばDF2を指定DFとする）の削除について説明する。入力部240を介してDF2の削除が指示されると、制御部210の制御によりディレクトリ削除コマンドが生成され、リーダー・ライター230からコンタクト部130に対して、生成されたディレクトリ削除コマンドとファイル名（DF2）が送信される。コンタクト部130でディレクトリ削除コマンドとファイル名（DF2）が受信されると、制御部110の制御により、ディレクトリ削除コマンドと共に送られてきたファイル名（DF2）のファイルがデータメモリ124から削除される。つまり、指定ファイル（この場合はDF2）が削除されることにより、この指定ファイルの配下に位置づけられたサブファイル（この場合はEF2-1及びEF2-2）が削除されることになる。

【0041】

なお、具体的なDF2の削除の流れについてはこの後詳しく説明する。

【0042】

上記したディレクトリ削除コマンドについてさらに説明する。ディレクトリ削除コマンドは、所定のディレクトリ（例えばDF2/メインファイルと仮定）と、この所定のディレクトリの配下に位置づけられたサブディレクトリ（例えばEF2-1、EF2-2/サブファイルと仮定）の削除を行わせるとともに、削除されたディレクトリ（サブディレクトリを含む）が占有していたメモリ領域を開放させるためのコマンドである。このディレクトリ削除コマンドによるディレクトリ（サブディレクトリを含む）の削除及びメモリ領域の開放により、ICカード100から不要なアプリケーションを削除して、新たなアプリケーションを追加することができる。

【0043】

次に、図3～図11を参照して、具体的なDF2の削除の流れ、及び具体的なメモリ領域の開放について説明する。図3は、ディレクトリ削除コマンドによるディレクトリ削除処理の概要を示すフローチャートである。

【0044】

図3に示すように、ディレクトリ削除コマンドによるディレクトリ削除処理は、9段階のステージにより実現される。また、ディレクトリ削除処理が実行されている間、ステージナンバー（ステージ00などの各処理の実行記録）がデータメモリ124のバックアップ領域（図4～図10に示すbackup-EF又はbackup-ef）に確保される。さらに言うと、処理が進行する度に、つまり、ステージナンバーが更新される度に、更新されたステージナンバーがデータメモリ124のバックアップ領域に確保される。また、定義情報及び実データなどもバックアップ領域に確保される。さらに言うと、定義情報及び実データの再配置の際に、これら定義情報及び実データは、バックアップ領域に確保される。これにより、不慮の電源ダウン及びリセットなどが原因でディレクトリ削除処理が途中で中断されてしまった場合でも、正常復帰後に、ディレクトリ削除処理を再実行することが可能となる。

【0045】

続いて、図4～図10に示すメモリマップを参照しつつ、図3に示すフローチャートの各ステージについて説明する。また、説明を分かり易くするために、図4～図10では、図2に示すファイル構造のDF2の削除を一例として取り上げて説明する。つまり、ディレクトリ削除コマンドにより、図2に示すファイル構造のDF2が削除されるケースについ

10

20

30

40

50

て説明する。

【 0 0 4 6 】

なお、図 4 は、図 3 に示すフローチャートのステージ 0 0 及び 0 1 を説明するための図である。図 5 は、図 3 に示すフローチャートのステージ 0 2 を説明するための図である。図 6 は、図 3 に示すフローチャートのステージ 0 4 を説明するための図である。図 7 は、図 3 に示すフローチャートのステージ 8 1 を説明するための図である。図 8 は、図 3 に示すフローチャートのステージ 8 2 を説明するための図である。図 9 は、図 3 に示すフローチャートのステージ 8 4 を説明するための図である。図 1 0 は、図 3 に示すフローチャートのステージ 2 0 を説明するための図である。

【 0 0 4 7 】

また、図 1 1 には、ワーキングメモリ 1 2 6 の作業領域に格納されるデータの名称、データの容量（バイト数）、及びデータの内容の一覧を示す。この作業領域には、上位アドレスから順に、図 1 1 に示すデータが格納されるものとする。さらに、図 1 2 には、データメモリ 1 2 4 のバックアップ領域に格納されるデータの名称、データの容量（バイト数）、及びデータの内容の一覧を示す。このバックアップ領域には、上位アドレスから順に、図 1 2 に示すデータが格納されるものとする。またさらに、図 1 3 には、データメモリ 1 2 4 のシステム領域に格納されるデータの名称、データの容量（バイト数）、及びデータの内容の一覧を示す。

【 0 0 4 8 】

ステージ 0 0 : コマンド実行準備

IC カード 1 0 0 の制御部 1 1 0 により、ディレクトリ削除コマンドが認識されると、このディレクトリ削除コマンドの実行準備が行われる。この実行準備の一つとして、図 4 に示すアドレス X 0 R がポインタに格納される。アドレス X 0 R は、D F 2 の定義情報の先頭アドレスを示すものである。ポインタとは、削除対象及び削除対象外のファイル（E F など）のアドレス、及び削除対象及び削除対象外のファイル（D F、E F）の定義情報のアドレスを格納するものであり、メモリ 2 2 0 のバックアップ領域に設けられている（図 1 1 参照）。

【 0 0 4 9 】

ステージ 0 1 : ポインタの設定

このステージ 0 1 では、図 4 に示すように、アドレス X 1 R、アドレス X 2 R、アドレス Y 2 T O P、アドレス P Y 2 R、アドレス P Y 1 R、及びアドレス Y 0 R がポインタに格納される。アドレス X 1 R は、E F 2 - 1 の定義情報の先頭アドレスを示すものである。アドレス X 2 R は、E F 1 - 1 の定義情報の先頭アドレスを示すものである。アドレス Y 2 T O P は、e f 1 - 1（アプリケーションファイル）の先頭アドレスを示すものである。アドレス P Y 2 R は、e f 1 - 1（アプリケーションファイル）の一部の先頭アドレスを示すものである。アドレス P Y 1 R は、e f 2 - 1（アプリケーションファイル）の一部の先頭アドレスを示すものである。アドレス Y 0 R は、b a c k u p - e f の先頭アドレスを示すものである。

【 0 0 5 0 】

ステージ 0 2 : e f（実データ）の移動

このステージ 0 2 では、図 5 に示すように、e f 2 - 1 が削除され、e f 1 - 1 が移動され、アドレスが更新される。因みに、e f 1 - 1 は、分割移動される（図 4 参照）。

【 0 0 5 1 】

ステージ 0 4 : E F（定義情報）の移動

このステージ 0 4 では、図 6 に示すように、e f 2 - 1 の削除に伴い、D F 2 及び E F 2 - 1 が削除され、D F 1 - 1 及び E F 1 - 1 が移動され、アドレスが更新される。

【 0 0 5 2 】

ステージ 8 1 : ポインタの設定

このステージ 8 1 では、図 7 に示すようなアドレスがポインタに格納される。

【 0 0 5 3 】

10

20

30

40

50

ステージ 8 2 : e f (実データ) の移動

このステージ 8 2 では、図 8 に示すように、e f 2 - 2 が削除され、e f 1 - 1 - 1 が移動され、アドレスが更新される。因みに、e f 1 - 1 - 1 は、分割移動される。

【 0 0 5 4 】

ステージ 8 4 : E F (定義情報) の移動

このステージ 0 4 では、図 9 に示すように、e f 2 - 2 の削除に伴い、E F 2 - 2 が削除され、E F 1 - 1 - 1 が移動され、アドレスが更新される。

【 0 0 5 5 】

ステージ 2 0 : メモリ使用サイズ変更 & メモリチェック

このステージでは、図 1 0 に示すように、空き領域が初期化される (メモリ領域の開放) 。これにより、新たな記憶領域を確保することができる。

10

【 0 0 5 6 】

また、制御部 1 1 0 の制御により、図 4 に示すような配置が実現されている。すなわち、定義情報 (D F 、 E F) が格納されるときには、データメモリ 1 2 4 の記憶領域の一端から順に定義情報が配置され、実データ (e f) が格納されるときには、データメモリ 1 2 4 の記憶領域の他端から順に実データが配置される。つまり、制御部 1 1 0 は、配置手段として機能している。

【 0 0 5 7 】

さらに、制御部 1 1 0 の制御により、定義情報の削除に伴い記憶領域の一端から順に定義情報が再配置され、実データの削除に伴い記憶領域の他端から順に実データが再配置される。つまり、制御部 1 1 0 は、格納位置制御手段として機能している。

20

【 0 0 5 8 】

以上説明したように、制御部 1 1 0 の制御により、図 4 ~ 図 1 0 に示すように、定義情報及び実データの格納位置が制御されている。このような定義情報及び実データの配置及び再配置を実現するために、定義情報及び実データが配置されている位置を示すアドレスデータが、常時、管理され、バックアップ領域にバックアップされている。さらに言うと、これらアドレスデータは、定義情報及び実データの再配置に伴い更新され、この更新のタイミングでバックアップ領域にバックアップされている。

【 0 0 5 9 】

なお、図 3 に示すフローチャートでは、実データと定義情報が交互に削除され再配置されるケースを例にとり説明したが、この発明はこれに限定されるものではない。例えば、削除対象となる実データを全て削除して再配置してから、削除対象となる定義情報を全て削除して再配置するようにしてもよい。

30

【 0 0 6 0 】

次に、図 1 4 ~ 図 1 6 を参照して、データ更新方法をさらに具体的に説明する。

【 0 0 6 1 】

ディレクトリ削除コマンドによるディレクトリ削除処理において、データメモリ 1 2 4 のバックアップ領域にデータを書き込むときに、メモリセルに割り当てられるフラグ (F L A G 1 及び F L A G 2) が使用される。このフラグは、コマンド実行中の電源ダウンに備え、コマンド処理履歴を残すものである。特に、データ書き換え中に電源ダウンが起こった場合、データは補償されないため、十分な考慮を要する。

40

【 0 0 6 2 】

フラグデータが \$ 8 0 のときフラグは 1 を示し、フラグデータが \$ 0 0 のときフラグは 0 を示すものとする。ところが、上記したように、フラグ書き換え中に電源電圧の低下又はリセットが生じた場合、フラグデータがどのようなデータになるか分からない。従って、フラグデータが \$ 8 0 以外のときは、フラグは 0 を示すものと見なす。

【 0 0 6 3 】

ディレクトリ削除コマンドによるディレクトリ削除処理では、図 1 4 に示すように、二つのフラグとデータバックアップ領域を使用し、データの更新が行われる。この手法により、更新前のデータ及び更新後のデータのうちの少なくとも一方が、補償されることになる

50

。

【 0 0 6 4 】

ここで、図 1 5 及び図 1 6 を参照して、データ更新について説明する。

【 0 0 6 5 】

一例として、データ列 A をデータ列 B に更新する場合について説明する。最初に、フラグデータのチェックが行われる。因みに、初期状態では両フラグ (F L A G 1 及び F L A G 2) は 0 を示すものとする。両フラグが 0 を示していない場合には、エラーとなる。エラーの場合、図 1 5 及び図 1 6 に示すように、両フラグが示す値に応じて、処理が継続される。つまり、両フラグが 0 を示していないのは、前回のコマンドの処理が中断していることである。

10

【 0 0 6 6 】

初期状態で両フラグが 0 を示している場合には、作業領域にて、データ列 B が作成され、データ列 B がバックアップ領域に保存される。これに伴い、 F L A G 1 = 1 となる。続いて、データ列 B がデータ格納領域に保存される。これに伴い、 F L A G 2 = 1 となる。この後、 F L A G 1 = 0 、 F L A G 2 = 0 となり、初期状態に戻される。

【 0 0 6 7 】

次に、図 1 7 ~ 図 3 6 に示すフローチャートを参照して、図 3 に示す各ステージをより詳細に説明する。なお、図中の略称の示す内容は、以下の通りである。

【 0 0 6 8 】

W T X : 待ち時間延長

20

W D L E N : 定義情報の構成単位 (D F 及び E F の定義情報は一定の長さで構成される)

E r r o r : コマンド中断とエラーレスポンスの出力

\$: 1 6 進数表現 (\$ 8 0 = 1 2 8 (1 0 進数表現))

まず、図 1 7 及び図 1 8 に示すフローチャートを参照して、図 3 に示すフローチャートのステージ 0 0 を説明する。

【 0 0 6 9 】

受信電文が W T X 応答である場合、 W T X 応答処理ルーチン (S T X - 1) にジャンプする。受信電文が W T X 応答でなく、データバックアップ用 E F が存在しなければ、エラーが出力される。受信電文が W T X 応答でなく、データバックアップ用 E F が存在する場合、データバックアップ用 E F のアドレス (トップアドレス) と、バックアップ領域のサイズが取得されワーキングメモリ 1 2 6 (名称 : G B U F F E F T O P 、 G B U F F L C) に保存される。

30

【 0 0 7 0 】

F L A G 0 が ' \$ 0 0 ' である場合、前回のコマンド処理が中断されており、コマンド異常終了処理ルーチン (S T X - 0) にジャンプする。 F L A G 0 が ' \$ 0 0 ' でなく、カレント D F (指定された D F) が M F である場合、エラーが出力される。 F L A G 0 が ' \$ 0 0 ' でなく、カレント D F (指定された D F) が M F でない場合、削除する D F (カレント D F) のシーケンスナンバー、削除する D F のサイズ、および削除する D F の親 D F のシーケンスナンバー、が取得され、ワーキングメモリ 1 2 6 (名称は順に D E L D F S N 、 D E L P F S N 、 D E L F S I Z) に保存される。さらに、削除される D F のディレクトリ情報 (定義情報) のアドレスがワーキングメモリ 1 2 6 (名称 : X O R) に保存される。なお、ここでフローチャートの説明は、フローチャートの ' A ' へ移行する。

40

【 0 0 7 1 】

カレント D F (指定 D F) に子 D F が存在する場合、エラーが出力される。カレント D F (指定 D F) に子 D F が存在しない場合、ステージナンバー ' \$ 0 1 ' がワーキングメモリ 1 2 6 (G S T G R) に保存される。さらに、 ' \$ 0 1 ' がワーキングメモリ 1 2 6 (G C O N T R : W T X 処理カウンタ) に設定される。そして、ポインタの更新ルーチンがコールされ、設定されたワーキングメモリ 1 2 6 (G S T G R 、 X O R) のデータがデータメモリ 1 2 4 (順に G S T G E 、 X O E) に保存される。その後、フラグ F L A G 1 、 F L A G 2 がクリアされる (F L A G 1 、 2 = ' \$ 0 0 ') 。

50

【 0 0 7 2 】

さらに、ワーキングメモリ 1 2 6 のデータ (D E L D F S N、D E L P F S N、D E L F S I Z) がデータメモリ 1 2 4 (順に D E L D F S N E、D E L P F S N E、D E L F S I Z E) に保存される。書き込みが正常に行われない場合、エラーが出力される。書き込みが正常に行われた場合、F L A G 0 が ' \$ 8 0 ' にされる。この後、処理は、フローチャートのステージ 0 1 - 0 に進む。

【 0 0 7 3 】

続いて、図 1 9 及び図 2 0 に示すフローチャートを参照して、図 3 に示すフローチャートのステージ 0 1 又はステージ 8 1 を説明する。

【 0 0 7 4 】

F L A G 1、F L A G 2 がクリアされる (F L A G 1、2 = ' \$ 0 0 ')。ワーキングメモリ 1 2 6 (G S T G R) のステージナンバーが ' \$ 0 1 ' である場合、ワーキングメモリ 1 2 6 (Y 0 R) にデータ領域の最下位アドレス (D I R P 2) が設定され、ワーキングメモリ 1 2 6 (X 1 R) にワーキングメモリ 1 2 6 (X 0 R) のデータが設定される。さらに、削除する D F の定義情報から削除する E F の定義情報がアドレスの上位に向かって W D L E N 単位にサーチされ、そのアドレスがワーキングメモリ 1 2 6 (X 1 R) に保存される。このとき、メモリ空き領域の最下位アドレス (D I R P 1) までサーチされ、削除する E F が見つからない場合は、ステージ 0 2 - 3 にジャンプする。削除する E F が見つかった場合は、削除する E F の定義情報から E F のデータ領域が特定され、E F トップアドレスと E F サイズが足されたものがワーキングメモリ 1 2 6 (Y 0 R) に保存し直される。以上はワーキングメモリ 1 2 6 (G S T G R) のステージナンバーが ' \$ 0 1 ' の場合に行われる。

【 0 0 7 5 】

この後、アドレスの上位に向かって、削除しない E F の定義情報がサーチされ、アドレスがワーキングメモリ 1 2 6 (X 2 R) に保存される (ここでフローチャートの説明は、フローチャートの ' B ' へ移行する)。このとき、メモリ空き領域の最下位アドレス (D I R P 1) までサーチされ、削除しない (移動する) E F が見つからない場合、ステージ 0 2 - 3 にジャンプする。削除しない E F が見つかった場合、削除しない E F の定義情報から E F のデータ領域トップアドレスとサイズが特定される。そのサイズが、データバックアップ領域のサイズ以下であれば、削除しない E F のデータ領域トップアドレスがワーキングメモリ 1 2 6 (P Y 2 R) に保存され、E F のサイズがワーキングメモリ 1 2 6 (L C R) に保存され、ワーキングメモリ 1 2 6 (Y 0 R) から削除しない E F のサイズを引いたものがワーキングメモリ 1 2 6 (P Y 1 R) に保存される。

【 0 0 7 6 】

また、削除される E F のサイズが、データバックアップ領域のサイズより大きければ、削除される E F のサイズをデータバックアップ領域のサイズで割った余りがワーキングメモリ 1 2 6 (L C R) に保存され、ワーキングメモリ 1 2 6 (Y 0 R) からワーキングメモリ 1 2 6 (L C R) のデータが引かれたものがワーキングメモリ 1 2 6 (P Y 1 R) に保存され、削除しない E F のトップアドレスとサイズが足されたものからワーキングメモリ 1 2 6 (L C R) のデータが引かれたものが、ワーキングメモリ 1 2 6 (P Y 2 R) に保存される。

【 0 0 7 7 】

従って、ワーキングメモリ 1 2 6 (Y 0 R、X 1 R、X 2 R、P Y 1 R、P Y 2 R、L C R) のデータが設定され、ワーキングメモリ 1 2 6 (G S T G R) のステージナンバーが 1 インクリメントされる。そして、ポインタの更新ルーチンがコールされ、ワーキングメモリ 1 2 6 (G S T G R、Y 0 R、X 1 R、X 2 R、P Y 1 R、P Y 2 R、L C R) のデータがデータメモリ 1 2 4 (G S T G E、Y 0 E、X 1 E、X 2 E、P Y 1 E、P Y 2 E、L C E) に保存される。この後、処理は、フローチャートの S T 0 2 - 0 へジャンプする。

【 0 0 7 8 】

10

20

30

40

50

続いて、図 2 1 及び図 2 2 に示すフローチャートを参照して、図 3 に示すフローチャートのステージ 0 2 又はステージ 8 2 を説明する。

【 0 0 7 9 】

ワーキングメモリ 1 2 6 (P Y 2 R) に保存されるアドレスより、ワーキングメモリ 1 2 6 (L C R) に示されるバイト数分のデータが、バックアップ領域に保存される。そして、 F L A G 1 がクリアされ (F L A G 1 = ' \$ 0 0 ')、処理がステージ 0 2 - 1 に進む。

【 0 0 8 0 】

ステージ 0 2 - 1 では、バックアップされたデータが、ワーキングメモリ 1 2 6 (P Y 1 R) に保存される。そして、 F L A G 2 がクリアされ (F L A G 2 = ' \$ 0 0 ')、処理がステージ 0 2 - 2 に進む。

【 0 0 8 1 】

ステージ 0 2 - 2 では、ワーキングメモリ 1 2 6 (P Y 2 R) のデータが、移動した E F のトップアドレスに等しくない場合、すなわち削除しない E F のデータ領域がすべて移動されていない場合、ワーキングメモリ 1 2 6 (L C R) にバックアップ領域サイズが保存され、ワーキングメモリ 1 2 6 (P Y 1 R、P Y 2 R) のデータがともにバックアップ領域サイズ部だけ減じられる。そして、ポインタの更新ルーチンがコールされ、ワーキングメモリ 1 2 6 (G S T G R、Y 0 R、X 1 R、X 2 R、P Y 1 R、P Y 2 R、L C R) のデータがデータメモリ 1 2 4 (G S T G E、Y 0 E、X 1 E、X 2 E、P Y 1 E、P Y 2 E、L C E) に保存される。そして、処理が、ステージ 0 2 - 0 にジャンプする。

【 0 0 8 2 】

ワーキングメモリ 1 2 6 (P Y 2 R) のデータが、移動した E F のトップアドレスに等しい場合、すなわち削除しない E F のデータ領域がすべて移動された場合、ワーキングメモリ 1 2 6 (Y 0 R) のデータがワーキングメモリ 1 2 6 (P Y 1 R) のデータに変更される。ワーキングメモリ 1 2 6 (G S T G R) のステージナンバーが ' \$ 8 1 ' または ' \$ 8 2 ' である場合、ワーキングメモリ 1 2 6 (G S T G R) が ' \$ 8 4 ' にされる。ワーキングメモリ 1 2 6 (G S T G R) のステージナンバーが、 ' \$ 8 1 ' または ' \$ 8 2 ' でない場合、ワーキングメモリ 1 2 6 (X 1 R) のデータが、ワーキングメモリ 1 2 6 (X 0 R) のデータと削除するディレクトリ定義情報のサイズ分 (W D L E N × 2) との加算値とされ、ワーキングメモリ 1 2 6 (G S T G R) が ' \$ 0 4 ' にされる。そして、ポインタの更新ルーチンがコールされ、ワーキングメモリ 1 2 6 (G S T G R、Y 0 R、X 1 R、X 2 R、P Y 1 R、P Y 2 R、L C R) のデータが、データメモリ 1 2 4 (G S T G E、Y 0 E、X 1 E、X 2 E、P Y 1 E、P Y 2 E、L C E) に保存される。その後、処理は、ステージ 0 4 - 0 にジャンプする。

【 0 0 8 3 】

ステージ 0 2 - 3 では、ワーキングメモリ 1 2 6 (X 2 R) にメモリ空き領域の最下位アドレス (D I R P 1) が保存される。ワーキングメモリ 1 2 6 (Y 0 R) のデータがワーキングメモリ 1 2 6 (P Y 1 R) のデータに変更される。ワーキングメモリ 1 2 6 (G S T G R) のステージナンバーが ' \$ 8 1 ' または ' \$ 8 2 ' である場合、ワーキングメモリ 1 2 6 (G S T G R) が ' \$ 8 4 ' にされる。ワーキングメモリ 1 2 6 (G S T G R) のステージナンバーが、 ' \$ 8 1 ' または ' \$ 8 2 ' でない場合、ワーキングメモリ 1 2 6 (X 1 R) のデータが、ワーキングメモリ 1 2 6 (X 0 R) のデータと削除するディレクトリ定義情報のサイズ分 (W D L E N × 2) との加算値とされ、ワーキングメモリ 1 2 6 (G S T G R) が ' \$ 0 4 ' にされる。そして、ポインタの更新ルーチンがコールされ、ワーキングメモリ 1 2 6 (G S T G R、Y 0 R、X 1 R、X 2 R、P Y 1 R、P Y 2 R、L C R) のデータが、データメモリ 1 2 4 (G S T G E、Y 0 E、X 1 E、X 2 E、P Y 1 E、P Y 2 E、L C E) に保存される。その後、処理はステージ 0 4 - 0 にジャンプする。

【 0 0 8 4 】

続いて、図 2 3、図 2 4、及び図 2 5 のフローチャートを参照して、図 3 に示すフローチャート

10

20

30

40

50

ャートのステージ 0 4 又はステージ 8 4 を説明する。

【 0 0 8 5 】

F L A G 1、2 がクリアされる (F L A G = ' \$ 0 0 ')。ワーキングメモリ 1 2 6 (X 1 R) のデータとワーキングメモリ 1 2 6 (X 2 R) のデータとが等しい場合、処理はステージ 0 4 - 1 にジャンプする。ワーキングメモリ 1 2 6 (X 1 R) のデータとワーキングメモリ 1 2 6 (X 2 R) のデータとが等しくなく、ワーキングメモリ 1 2 6 (X 1 R) に保存されるアドレスからの定義情報が削除対象である場合、そのアドレスに定義情報サイズ分が加算され、ワーキングメモリ 1 2 6 (X 1 R) に保存し直される。

【 0 0 8 6 】

また、ワーキングメモリ 1 2 6 (X 1 R) に保存されるアドレスからの定義情報が削除対象でない場合、その定義情報がワーキングメモリ 1 2 6 (X 0 R) に保存されるアドレスに転送される。そして、ワーキングメモリ 1 2 6 (X 0 R、X 1 R) のアドレスに W D L E N (定義情報サイズ) が加算される。そして、ポインタの更新ルーチンがコールされ、ワーキングメモリ 1 2 6 (G S T G R、X 0 R、Y 0 R、X 1 R、X 2 R、P Y 1 R、P Y 2 R、L C R) のデータがデータメモリ 1 2 4 (G S T G E、X 0 E、Y 0 E、X 1 E、X 2 E、P Y 1 E、P Y 2 E、L C E) に保存される。その後、処理はステージ 0 4 - 0 の先頭に戻る。

【 0 0 8 7 】

ステージ 0 4 - 1 では、ワーキングメモリ 1 2 6 (X 2 R) のデータがメモリ空き領域の最下位アドレス (D I R P 1) である場合、処理はステージ 0 4 - 2 にジャンプする。ワーキングメモリ 1 2 6 (X 2 R) のデータがメモリ空き領域の最下位アドレス (D I R P 1) でない場合、ワーキングメモリ 1 2 6 (X 2 R) のアドレスからの定義情報において、データ領域トップアドレス情報がワーキングメモリ 1 2 6 (P Y 1 R) のデータに置き換えられ、ワーキングメモリ 1 2 6 (X 0 R) のアドレスに転送される。ワーキングメモリ 1 2 6 (X 0 R、X 1 R) のアドレスに定義情報サイズ分 (W D L E N) が加算され、ワーキングメモリ 1 2 6 (G S T G R) が ' \$ 8 1 ' にされる。そして、ポインタの更新ルーチンがコールされ、ワーキングメモリ 1 2 6 (G S T G R、X 0 R、Y 0 R、X 1 R、X 2 R、P Y 1 R、P Y 2 R、L C R) のデータがデータメモリ 1 2 4 (G S T G E、X 0 E、Y 0 E、X 1 E、X 2 E、P Y 1 E、P Y 2 E、L C E) に保存される。その後、処理はステージ 0 1 - 0 にジャンプする。

【 0 0 8 8 】

ステージ 0 4 - 2 では、削除される D F の親 D F が M F でなく、且つ、削除される D F の親 D F の使用サイズ情報が設定されていない場合、ワーキングメモリ 1 2 6 (X 1 R) に、削除される D F の親 D F の定義情報のアドレスが設定される。それ以外の場合は、ワーキングメモリ 1 2 6 (X 1 R) に M F の定義情報のアドレスが設定される。そして、ワーキングメモリ 1 2 6 (G S T G R) のステージナンバーが ' \$ 2 0 ' にされる。そして、ポインタの更新ルーチンがコールされ、ワーキングメモリ 1 2 6 (G S T G R、X 0 R、Y 0 R、X 1 R、X 2 R、P Y 1 R、P Y 2 R、L C R) のデータがデータメモリ 1 2 4 (G S T G E、X 0 E、Y 0 E、X 1 E、X 2 E、P Y 1 E、P Y 2 E、L C E) に保存される。その後、処理はステージ 2 0 - 0 に進む。

【 0 0 8 9 】

続いて、図 2 6、図 2 7、及び図 2 8 のフローチャートを参照して、図 3 に示すフローチャートのステージ 2 0 を説明する。

【 0 0 9 0 】

ワーキングメモリ 1 2 6 (X 1 R) のアドレスからの使用サイズ情報が、削除領域サイズを引いた値に置き換えられ、以上のデータがバックアップ領域に保存される。そして、F L A G 1 がクリアされる (F L A G 1 = ' \$ 0 0 ')。その後、処理はステージ 2 0 - 1 に進む。

【 0 0 9 1 】

ステージ 2 0 - 1 では、バックアップ領域のデータがワーキングメモリ 1 2 6 (X 1 R)

のアドレスに転送される（この後、処理はフローチャートの‘C’へジャンプする）。そして、FLAG2がクリアされ（FLAG2 = ‘\$00’）、削除領域がメモリ初期状態化される。一定領域のメモリが初期状態化される毎にWTX要求が行われる。ワーキングメモリ126（GSTGR）のステージナンバーが‘\$40’にされる。そして、ポインタの更新ルーチンがコールされ、ワーキングメモリ126（GSTGR、X0R、Y0R、X1R、X2R、PY1R、PY2R、LCR）のデータがデータメモリ124（GSTGE、X0E、Y0E、X1E、X2E、PY1E、PY2E、LCE）に保存される。その後、処理は、ステージ40に進む。

【0092】

続いて、図29のフローチャートを参照して、図3に示すフローチャートのステージ40を説明する。

10

【0093】

メモリ管理情報のメモリ空き領域最下位アドレス情報（DIRP1）がワーキングメモリ126（X0R）のデータに、メモリ管理情報のデータ領域最下位アドレス情報（DIRP2）がワーキングメモリ126（Y0R）のデータに置き換えられ、以上のデータがバックアップ領域に保存される。そして、FLAG1がクリアされる（FLAG1 = ‘\$00’）。その後、処理はステージ40-1に進む。

【0094】

ステージ40-1では、バックアップ領域のデータがメモリ管理情報領域に保存される。そして、FLAG2がクリアされる（FLAG2 = ‘\$00’）。その後、処理はステージ40-2に進む。

20

【0095】

ステージ40-2では、MF以外のDFにて獲得されたセキュリティステータス（照合状態）がクリアされる。そして、FLAG0がクリアされ（FLAG0 = ‘\$00’）、正常終了となる。

【0096】

続いて、図30及び図31に示すフローチャートを参照して、コマンド異常終了対応ルーチン及びWTX応答処理を説明する。

【0097】

データメモリ124（DELDFSNE、DELPFSNE、DELF SIZE、GSTGE、X0E、Y0E、X1E、X2E、PY1E、PY2E、LCE）のデータが、ワーキングメモリ126（DELDFS N、DELPFS N、DELF SIZ、GSTGR、X0R、Y0R、X1R、X2R、PY1R、PY2R、LCR）に展開される。WTX処理カウンタの初期値である‘\$00’が、ワーキングメモリ126（GCONTR）に設定される。その後、処理はWTX応答処理ルーチン（STX-1）に進む。

30

【0098】

WTX応答処理ルーチンでは、FLAG1 = ‘\$80’かつFLAG2 = ‘\$00’の場合、ポインタの更新ルーチン（後半）がコールされ、処理はコマンド異常終了処理ルーチンにジャンプする。ワーキングメモリ126（GSTGR）のデータが‘\$01’または‘\$81’の場合、処理はステージ01-0にジャンプする。ワーキングメモリ126（GSTGR）のデータが‘\$04’または‘\$84’の場合、処理はステージ04-0にジャンプする（この後、処理はフローチャートの‘F’へジャンプする）。

40

【0099】

FLAG1 = ‘\$80’かつFLAG2 = ‘\$80’の場合において、ワーキングメモリ126（GSTGR）のデータが‘\$02’または‘\$82’の場合、処理はステージ02-0にジャンプし、ワーキングメモリ126（GSTGR）のデータが‘\$20’の場合、処理はステージ20-0にジャンプし、ワーキングメモリ126（GSTGR）のデータが‘\$40’の場合、処理はステージ40-0にジャンプする。

【0100】

FLAG1 = ‘\$00’かつFLAG2 = ‘\$80’であり、ワーキングメモリ126（

50

G S T G R) のデータが ' \$ 0 2 ' または ' \$ 8 2 ' の場合、ステージ 0 2 - 1 にジャンプし、ワーキングメモリ 1 2 6 (G S T G R) のデータが ' \$ 2 0 ' の場合、ステージ 2 0 - 1 にジャンプし、ワーキングメモリ 1 2 6 (G S T G R) のデータが ' \$ 4 0 ' の場合、ステージ 4 0 - 1 にジャンプする。

【 0 1 0 1 】

F L A G 1 = ' \$ 0 0 ' かつ F L A G 2 = ' \$ 0 0 ' であり、ワーキングメモリ 1 2 6 (G S T G R) のデータが ' \$ 2 0 ' の場合、ステージ 2 0 - 2 にジャンプし、 ' \$ 0 2 ' または ' \$ 8 2 ' の場合はステージ 0 2 - 2 にジャンプする。

【 0 1 0 2 】

続いて、図 3 2 に示すフローチャートを参照して、メモリ書き込みについて説明する。

10

【 0 1 0 3 】

図 3 2 のフローチャートに示すように、メモリ領域内の指定された書込先に、指定されたデータの書き込みを行う。このとき、電源電圧異常が検出された場合、エラーが出力される。電源電圧異常が検出されなければ、このサブルーチンにおける処理は終了となる。

【 0 1 0 4 】

続いて、図 3 3 に示すフローチャートを参照して、フラグのセット / クリアについて説明する。

【 0 1 0 5 】

図 3 3 のフローチャートに示すように、指定されたフラグに、指定されたデータの書き込みを行う。このとき、電源電圧異常が検出された場合、エラーが出力される。電源電圧異常が検出されなければ、このサブルーチンにおける処理は終了となる。

20

【 0 1 0 6 】

続いて、図 3 4 及び図 3 5 に示すフローチャートを参照して、ポインタの更新について説明する。

【 0 1 0 7 】

ワーキングメモリ 1 2 6 (G S T G R 、 X 0 R 、 Y 0 R 、 X 1 R 、 X 2 R 、 P Y 1 R 、 P Y 2 R 、 L C R) のデータが、ワーキングメモリ 1 2 6 (G B U F F R) に転送され、バックアップ領域に保存される。そして、F L A G 1 が ' \$ 8 0 ' にされ、処理はポインタの更新ルーチン (後半) にジャンプする。

【 0 1 0 8 】

ポインタの更新ルーチン (後半) では、バックアップ領域のデータがデータメモリ 1 2 4 (G S T G E) のアドレスから保存される。そして、F L A G 2 が ' \$ 8 0 ' にされる。ワーキングメモリ 1 2 6 (G S T G R) のデータが ' \$ 0 2 ' または ' \$ 8 2 ' である場合、ワーキングメモリ 1 2 6 (L C R) データが 6 4 で乗算され、1 加算され、2 倍されたものがワーキングメモリ 1 2 6 (G C O N T R) のデータに加算される。ワーキングメモリ 1 2 6 (G S T G R) のステージナンバーが ' \$ 0 2 ' または ' \$ 8 2 ' でない場合、ワーキングメモリ 1 2 6 (G C O N T R) のデータが 1 インクリメントされる (この後、処理はフローチャート ' G ' ヘジャンプ) 。ワーキングメモリ 1 2 6 (G C O N T R) のデータが ' \$ 1 2 ' を超えた場合、ワーキングメモリ 1 2 6 (G C O N T R) のデータが初期値 ' \$ 0 0 ' に戻され、W T X 要求が行われる。ワーキングメモリ 1 2 6 (G C O N T R) のデータが ' \$ 1 2 ' を超えない場合、処理は本ルーチンをコールした元のルーチンに戻る。

30

40

【 0 1 0 9 】

続いて、図 3 6 に示すフローチャートを参照して、メモリ初期状態化について説明する。

【 0 1 1 0 】

図 3 6 のフローチャートに示すように、指定されたスタートアドレスから、ある一定サイズのメモリの初期状態化が行われる。このとき、電源電圧異常が検出された場合、エラーが出力される。電源電圧異常が検出されずに、指定されたエンドアドレスまでメモリ初期状態化されていない場合、処理は W T X 要求へ移行する。電源電圧異常が検出されずに、指定されたエンドアドレスまでメモリ初期状態化されている場合、このサブルーチンにお

50

ける処理は終了となる。

【 0 1 1 1 】

次に、図 3 7 を参照して、フラグと実行ステータスについて説明する。

【 0 1 1 2 】

フラグ (F L A G 0 、 1 、 2) は、図 3 7 に示すように、書き換えられるようになっている。つまり、ディレクトリ削除コマンドに従った処理の実行中を示すフラグ (F L A G 0) は、処理の最初と最後に書き換えられる。また、このフラグ (F L A G 0) 以外のフラグ (F L A G 1 、 2) により、ポイントのバックアップ及びデータ領域のバックアップが行われている。なお、図 3 7 中の ' X X ' は、任意のデータを示す。フラグデータ ' \$ 8 0 ' 以外は、 ' \$ 0 0 ' として扱われる。

10

【 0 1 1 3 】

次に、D F シーケンスナンバーについて説明する。

【 0 1 1 4 】

D F シーケンスナンバーは、D F を識別するためのものであり、一つ一つの D F にユニークに割り当てられるものである。従来、この D F シーケンスナンバーは、1 バイト (= 2 5 6 通り) で管理されており、D F が創成されるたびにインクリメントされていた。ところが、ディレクトリ削除コマンドをサポートするには、このような従来の方法では、不十分である。つまり、D F の削除と追加が繰り返されるうちに、やがては同じシーケンスナンバーが割り当てられてしまう。

20

【 0 1 1 5 】

そこで、この発明では、シーケンスナンバーをビットマップ管理する事により、M F 以外に 3 2 個の D F の追加が可能となる。例えば、メモリ管理情報の 4 バイト (D F S E Q N O) にて管理する。

【 0 1 1 6 】

例えば、D F シーケンスナンバー 1、ナンバー 4、ナンバー 9 が使用されている場合、ビットマップは、以下のようになる。

【 0 1 1 7 】

00000000 00000000 00000001 00001001

上記状態から、D F シーケンスナンバー 9 のディレクトリが削除されると、ビットマップは、以下のようになる。

30

【 0 1 1 8 】

00000000 00000000 00000000 00001001

上記状態から、一つディレクトリが創成されると、D F シーケンスナンバー 1、ナンバー 2、ナンバー 4 が使用されることになり (D F シーケンスナンバーの小さいものから順に使用される場合)、ビットマップは、以下のようになる。

【 0 1 1 9 】

00000000 00000000 00000000 00001011

上記したビットマップは、制御部 1 1 0 により生成され、データメモリ 1 2 4 に記憶される。つまり制御部 1 1 0 は、ビットマップ生成手段として機能する。データメモリ 1 2 4 は、ビットマップデータを格納するビットマップデータ格納手段として機能する。

40

【 0 1 2 0 】

次に、W T X について説明する。

【 0 1 2 1 】

I C カード処理装置 2 0 0 から I C カード 1 0 0 に対して、コマンド電文が送信されるとする。このとき、I C カードでは、受信したコマンドの指定する処理が実行される。そして、I C カードから I C カード処理装置に対して、処理結果を示すレスポンスが返信される。I C カード 1 0 0 と I C カード処理装置 2 0 0 は、このような一連の電文のやり取りを行っている。

【 0 1 2 2 】

I C カード 1 0 0 でコマンド電文が受信されたとき、この受信されたコマンド電文に相当

50

するコマンド処理が、ある一定時間（BWT）より長い時間を必要とするとき、ICカード100はICカード処理装置200に対してWTX要求（待ち時間延長要求）を返信する。これに対し、ICカード処理装置200はICカード100に対して、WTX応答（待ち時間延長応答）を返信する。

【0123】

次に、図38～図40を参照して、DF定義情報、MF定義情報、及びEF定義情報について説明する。まず、図38～図40中の略称を説明する。

【0124】

WAI：定義情報の識別情報。

【0125】

DFSN：DFシーケンスナンバー

PFSN：親DFシーケンスナンバー

FSIZ：DFに割り当てられたメモリサイズ。FSIZ = 0のときサイズ管理は行わない。

【0126】

USIZ：DFの配下にあるDF及びEFの使用総サイズ。

【0127】

EFTOP：EFの先頭アドレス

EFsiz：EFに割り当てられた使用サイズ

図38に示すように、DF定義情報（長さ：WDLLEN×2）には、WAI、PFSN、（その他の情報）、DFSN、（DF名、その他の情報）、WAI、DFSN、（その他の情報）、FSIZ、USIZ、（その他の情報）などが含まれている。

【0128】

図39に示すように、MF定義情報（長さ：WDLLEN）には、WAI、DFSN、（その他の情報）、FSIZ、USIZ、（その他の情報）などが含まれている。因みに、このMF定義情報は、DF定義情報の後半部分と同じフォーマットである。

【0129】

図40に示すように、EF定義情報（長さ：WDLLEN）には、WAI、DFSN、（その他の情報）、EFTOP、EFsiz、（その他の情報）などが含まれている。

【0130】

続いて、アクセスキーによりアクセスが制限されたファイルの削除について説明する。

【0131】

例えば、図4のEF1-1のアクセスに必要なアクセスキーが図4のEF2-1に保持されているとする。

【0132】

このような状況下では、EF1-1のアクセスの際に、入力部240から入力されたアクセスキーと、EF2-1に保持されたアクセスキーとが照合され、これら両アクセスキーが一致したときに、EF1-1へのアクセスが許可される。上記したアクセスキーの照合及びアクセスの許可判定は、制御部110により行われる。

【0133】

また、このような状況下で、ディレクトリ削除コマンドによるディレクトリ削除処理によりEF2-1が削除された場合には、制御部110の制御により、EF1-2へのアクセスが禁止される。つまり、制御部110は、ファイルアクセス管理手段として機能する。これにより、アクセスキーによりアクセスが制限されたファイルの情報の漏洩が防止される。

【0134】

【発明の効果】

この発明によれば、新たなアプリケーションの設定の自由度に優れたICカード、ICカード処理システム、ICカード処理方法を提供できる。

【図面の簡単な説明】

10

20

30

40

50

【図 1】この発明の IC カード処理システムの概略構成を示すブロック図である。

【図 2】図 1 に示す IC カードのデータメモリに構築されるファイル構造の概要を説明するための図である。

【図 3】ディレクトリ削除コマンドによるディレクトリ削除処理の全体の概要を示すフローチャートである。

【図 4】図 3 に示すフローチャートのステージ 0 0 及び 0 1 を説明するためのメモリマップを示す図である。

【図 5】図 3 に示すフローチャートのステージ 0 2 を説明するためのメモリマップを示す図である。

【図 6】図 3 に示すフローチャートのステージ 0 4 を説明するためのメモリマップを示す図である。 10

【図 7】図 3 に示すフローチャートのステージ 8 1 を説明するためのメモリマップを示す図である。

【図 8】図 3 に示すフローチャートのステージ 8 2 を説明するためのメモリマップを示す図である。

【図 9】図 3 に示すフローチャートのステージ 8 4 を説明するためのメモリマップを示す図である。

【図 1 0】図 3 に示すフローチャートのステージ 2 0 を説明するためのメモリマップを示す図である。

【図 1 1】ワーキングメモリの作業領域に格納されるデータの名称、データの容量（バイト数）、及びデータの内容の一覧を示す図である。 20

【図 1 2】データメモリのバックアップ領域に格納されるデータの名称、データの容量（バイト数）、及びデータの内容の一覧を示す図である。

【図 1 3】データメモリのシステム領域に格納されるデータの名称、データの容量（バイト数）、及びデータの内容の一覧を示す図である。

【図 1 4】データ更新処理を説明するためのメモリマップを示す図である。

【図 1 5】データ更新処理を説明するためのフローチャートである。

【図 1 6】データ更新処理を説明するための、図 1 5 に示すフローチャートから続くフローチャートである。

【図 1 7】図 3 に示すフローチャートのステージ 0 0 を説明するためのフローチャートである。 30

【図 1 8】図 3 に示すフローチャートのステージ 0 0 を説明するための、図 1 7 に示すフローチャートから続くフローチャートである。

【図 1 9】図 3 に示すフローチャートのステージ 0 1 又はステージ 8 1 を説明するためのフローチャートである。

【図 2 0】図 3 に示すフローチャートのステージ 0 1 又はステージ 8 1 を説明するための、図 1 9 に示すフローチャートから続くフローチャートである。

【図 2 1】図 3 に示すフローチャートのステージ 0 2 又はステージ 8 2 を説明するためのフローチャートである。

【図 2 2】図 3 に示すフローチャートのステージ 0 2 又はステージ 8 2 を説明するための、図 2 1 に示すフローチャートから続くフローチャートである。 40

【図 2 3】図 3 に示すフローチャートのステージ 0 4 又はステージ 8 4 を説明するためのフローチャートである。

【図 2 4】図 3 に示すフローチャートのステージ 0 4 又はステージ 8 4 を説明するための、図 2 3 に示すフローチャートから続くフローチャートである。

【図 2 5】図 3 に示すフローチャートのステージ 0 4 又はステージ 8 4 を説明するための、図 2 4 に示すフローチャートから続くフローチャートである。

【図 2 6】図 3 に示すフローチャートのステージ 2 0 を説明するためのフローチャートである。

【図 2 7】図 3 に示すフローチャートのステージ 2 0 を説明するための、図 2 6 に示すフ 50

ローチャートから続くフローチャートである。

【図 28】図 3 に示すフローチャートのステージ 20 を説明するための、図 27 に示すフローチャートから続くフローチャートである。

【図 29】図 3 に示すフローチャートのステージ 40 を説明するためのフローチャートである。

【図 30】コマンド異常終了対応ルーチン及び W T X 応答処理を説明するためのフローチャートである。

【図 31】コマンド異常終了対応ルーチン及び W T X 応答処理を説明するための、図 30 に示すフローチャートから続くフローチャートである。

【図 32】メモリ書き込みについて説明するためのフローチャートである。

10

【図 33】フラグのセット / クリアについて説明するためのフローチャートである。

【図 34】ポインタの更新について説明するためのフローチャートである。

【図 35】ポインタの更新について説明するための、図 34 に示すフローチャートから続くフローチャートである。

【図 36】メモリ初期状態化について説明するためのフローチャートである。

【図 37】フラグデータと実行ステータスの関係を示す図である。

【図 38】D F 定義情報の一例を示す図である。

【図 39】M F 定義情報の一例を示す図である。

【図 40】E F 定義情報の一例を示す図である。

20

【符号の説明】

1 0 0 ... I C カード

1 1 0 ... 制御部

1 2 0 ... メモリ

1 2 2 ... プログラムメモリ

1 2 4 ... データメモリ

1 2 6 ... ワーキングメモリ

1 3 0 ... コンタクト部

2 0 0 ... I C カード処理装置

2 1 0 ... 制御部

2 2 0 ... メモリ

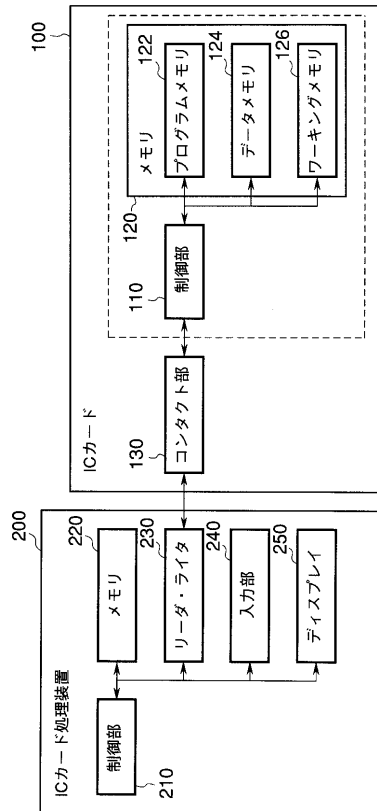
2 3 0 ... リーダ・ライター

2 4 0 ... 入力部

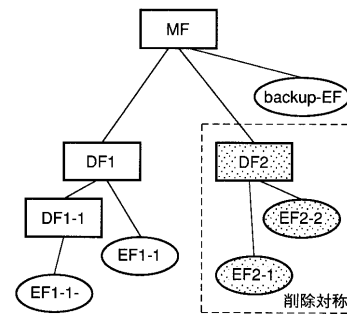
2 5 0 ... ディスプレイ

30

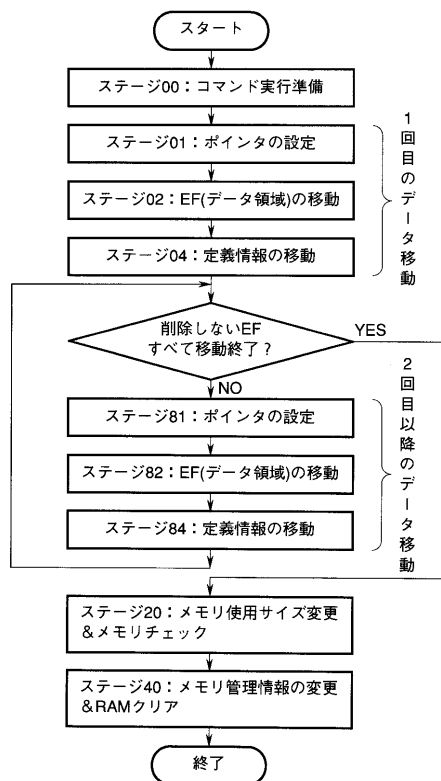
【図 1】



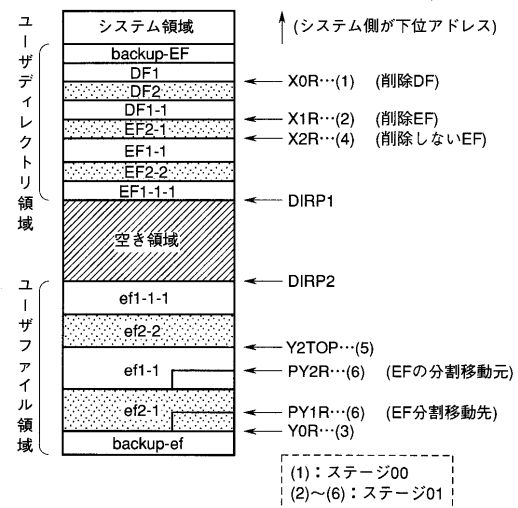
【図 2】



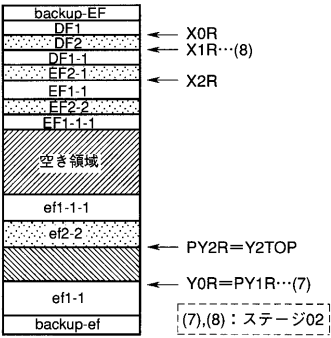
【図 3】



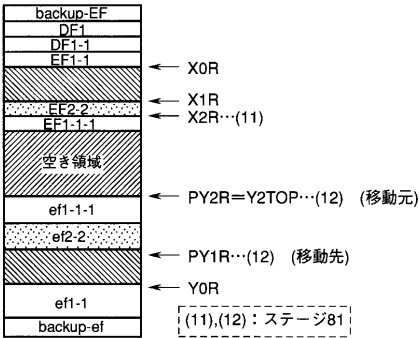
【図 4】



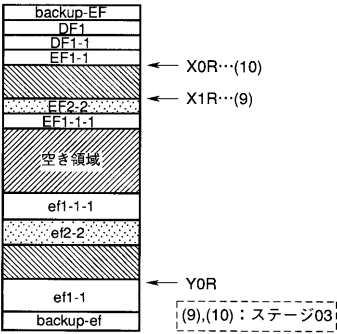
【図5】



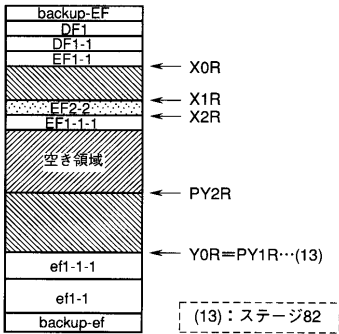
【図7】



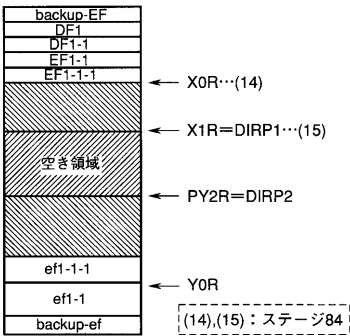
【図6】



【図8】



【図9】

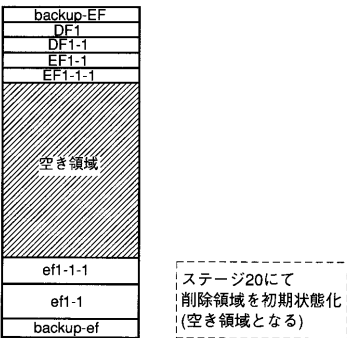


【図11】

(名称)	(バイト数)	(内容)
GBUFFEFTOP	2	データバッファ用EFのトップアドレス
GBUFFLC	2	上記EFのバックアップ領域(GUFFE)のサイズ
DELDFSN	1	削除対象ディレクトリのDF-No.
DELPFSN	1	削除対象ディレクトリのDF-No.
DELSIZ	2	削除対象ディレクトリの定義サイズ・情報
GSTGR	1	コマンドのステージNo.
X0R	2	ポイント: 移動し終えた定義情報のアドレス
Y0R	2	ポイント: 移動し終えたデータ領域のアドレス
X1R	2	ポイント: 削除するEF定義情報トップアドレス
X2R	2	ポイント: 削除しないEF定義情報トップアドレス
PY1R	2	ポイント: データ領域の分割移動先のアドレス
PY2R	2	ポイント: データ領域の分割移動元のアドレス
LCR	2	移動するデータのサイズ
Y2TOP	2	移動するEFのデータ領域トップアドレス
Y2SIZE	2	移動するEFのサイズ
GCONTR	1	移動するEFのサイズ
GBUFFR	WDLEN~255	WTX要求処理用カウンタ
		データ一時保存領域(EEPROM書き込み用)

※使用領域は、バッファサイズ(GBUFFR)+28バイト

【図10】

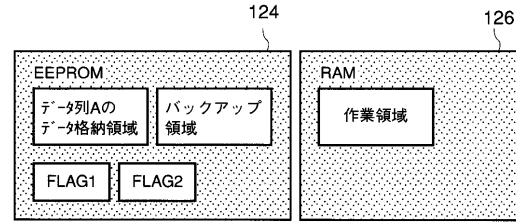


【図 1 2】

(名称)	(バイト数)	(内容)
FLAG0	1	コマンド処理中フラグ
FLAG1	1	データバックアップ完了フラグ
FLAG2	1	データ本更新完了フラグ
DELDIFSNE	1	削除対象ディレクトリのDF-No.
DELPFSNE	1	削除対象ディレクトリの親DF-No.
DELFSSIZE	2	削除対象ディレクトリの定義サイズ情報
GSTGE	1	コマンドのステージNo.
X0E	2	ホインタ: 移動し終えた定義情報のアドレス
Y0E	2	ホインタ: 移動し終えたデータ領域のアドレス
X1E	2	ホインタ: 削除するEF定義情報トップアドレス
X2E	2	ホインタ: 削除しないEF定義情報トップアドレス
PY1E	2	ホインタ: データ領域の分割移動先のアドレス
PY2E	2	ホインタ: データ領域の分割移動元のアドレス
LCE	2	移動するデータのサイズ
GBUFFE	WDLEN~255	データバックアップ領域(データ仮更新領域)

※使用領域は、バッファ領域サイズ(GBUFFE)+22バイト

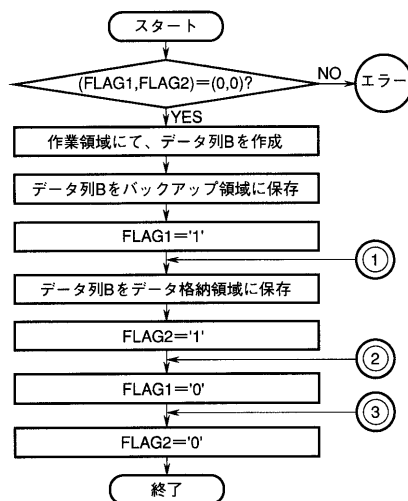
【図 1 4】



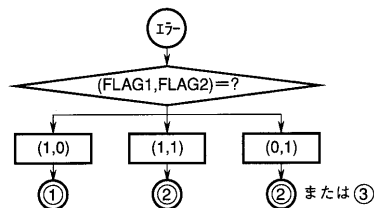
【図 1 3】

(名称)	(バイト数)	(内容)
DIRP1	2	メモリ空き領域の最下位アドレス
DIRP2	2	データ領域の最下位アドレス
DFSEQNO	4	DFシーケンスNo.(ビットマップ管理)

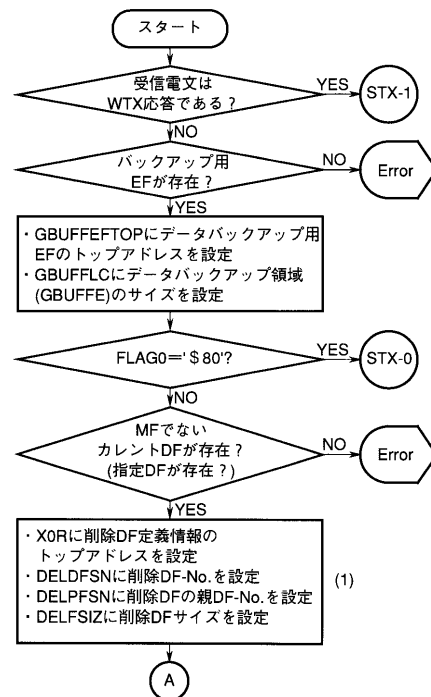
【図 1 5】



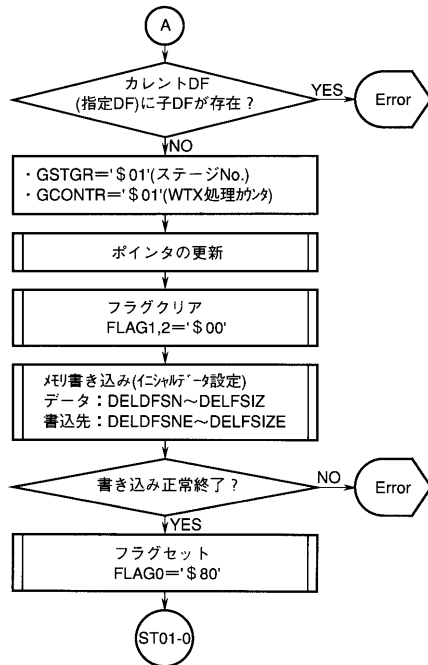
【図 1 6】



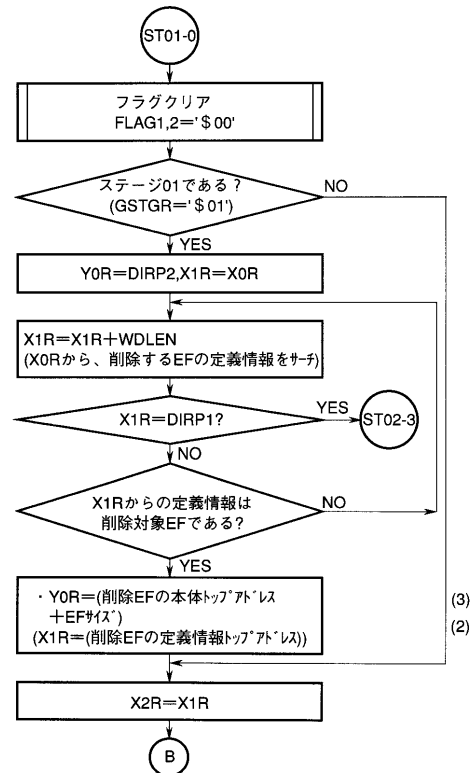
【図 1 7】



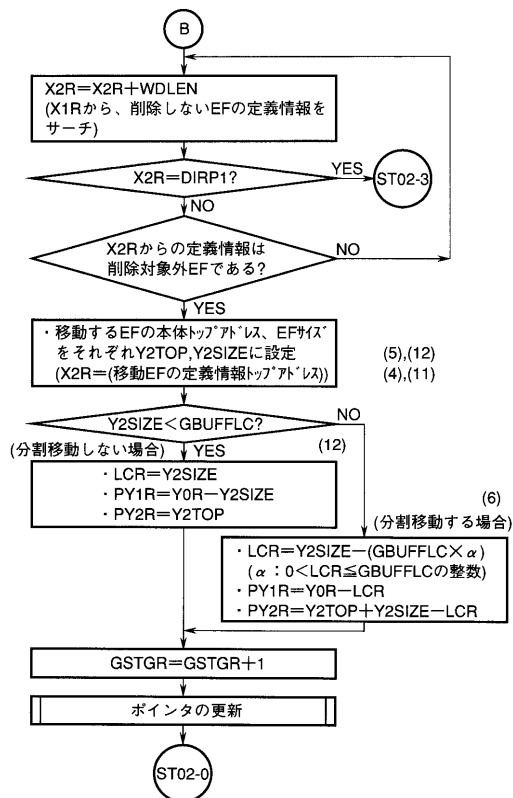
【図 18】



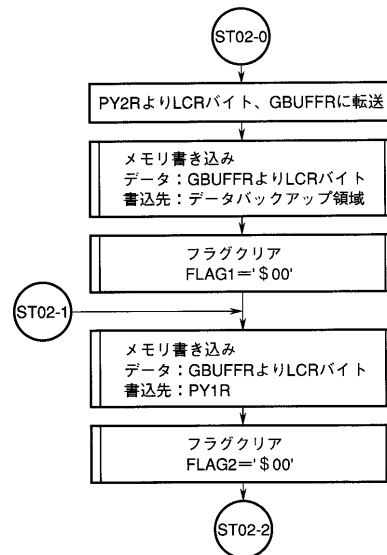
【図 19】



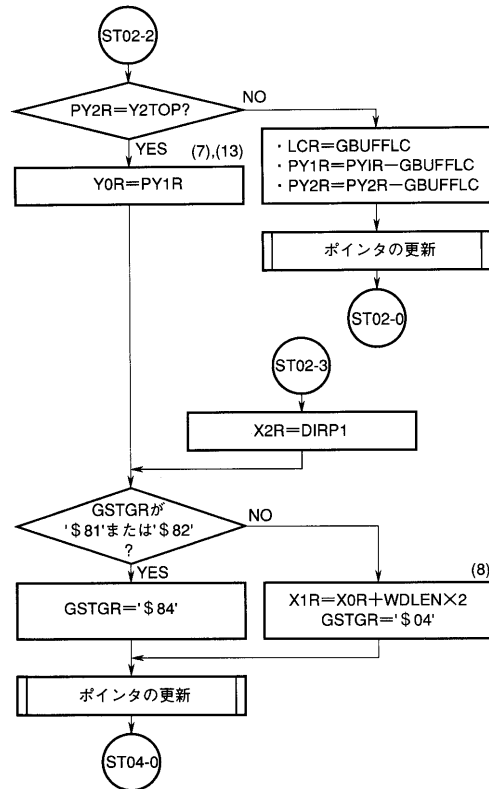
【図 20】



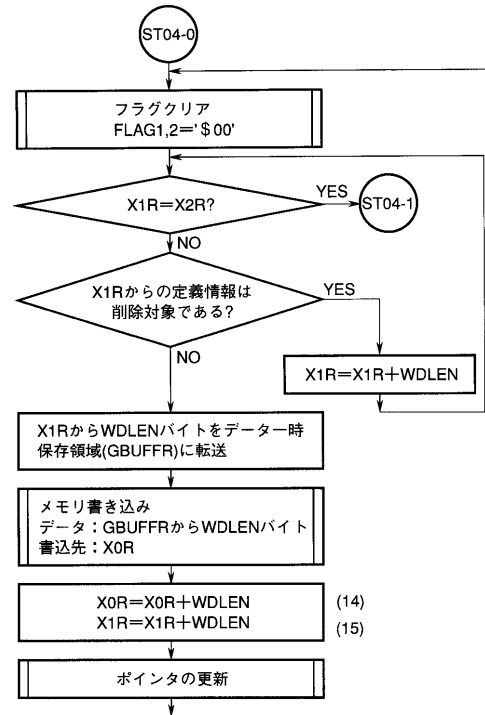
【図 21】



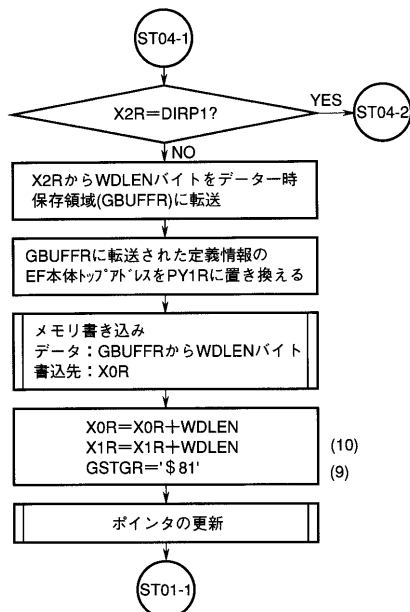
【図 22】



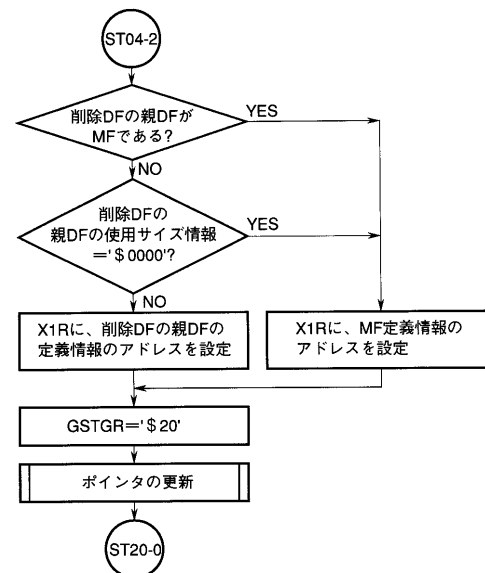
【図 23】



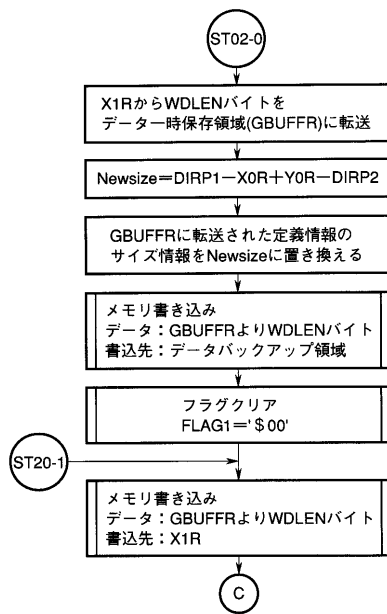
【図 24】



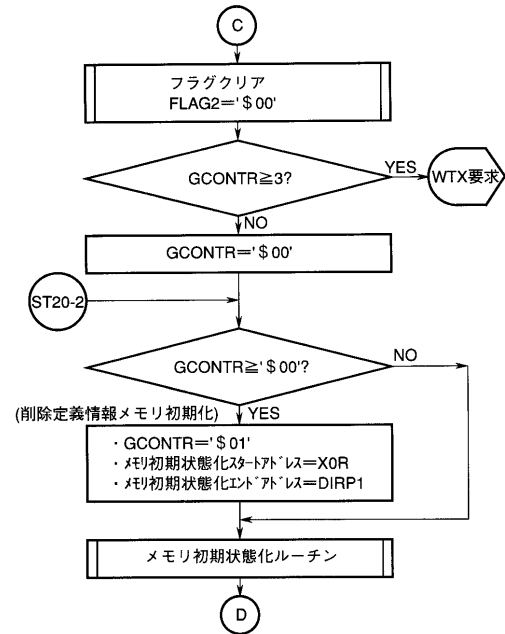
【図 25】



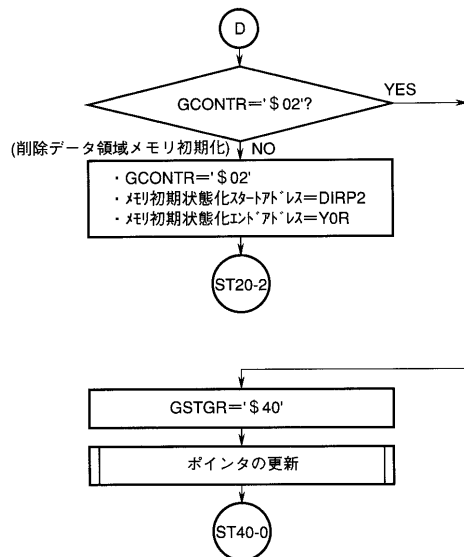
【図 26】



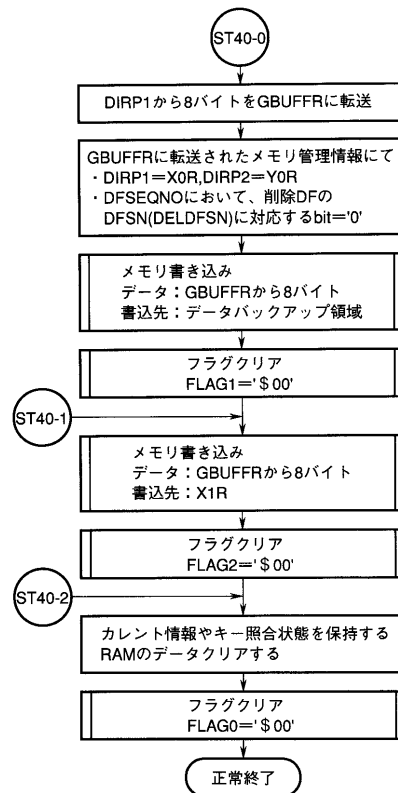
【図 27】



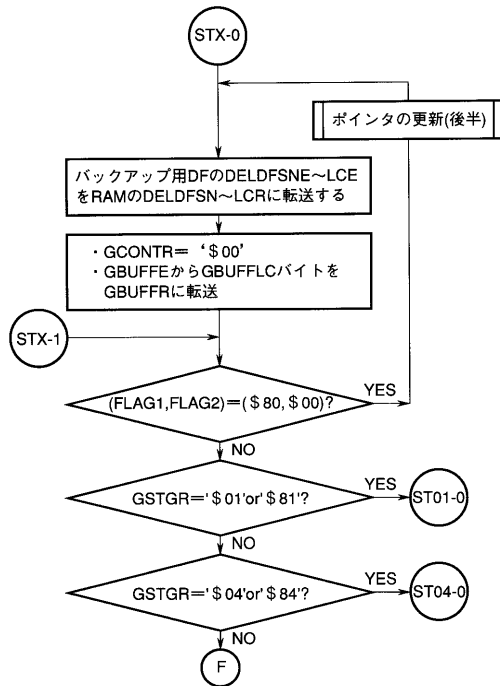
【図 28】



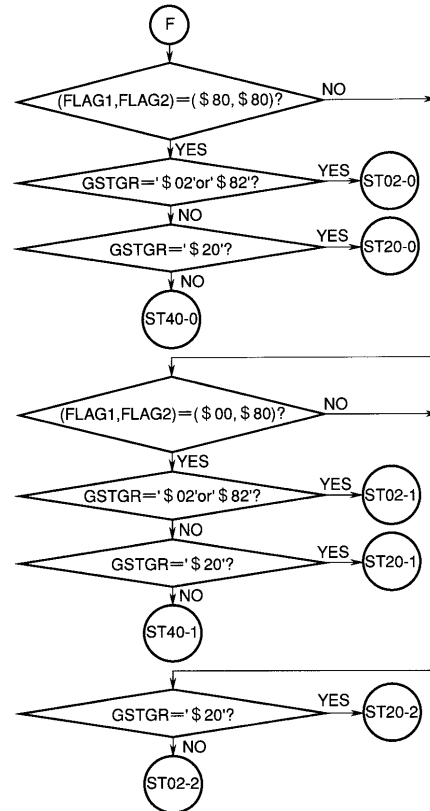
【図 29】



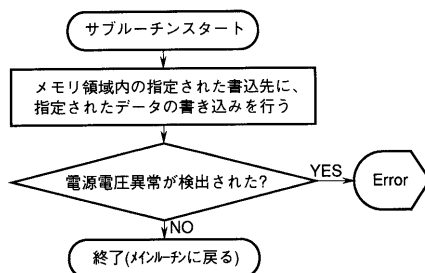
【図 30】



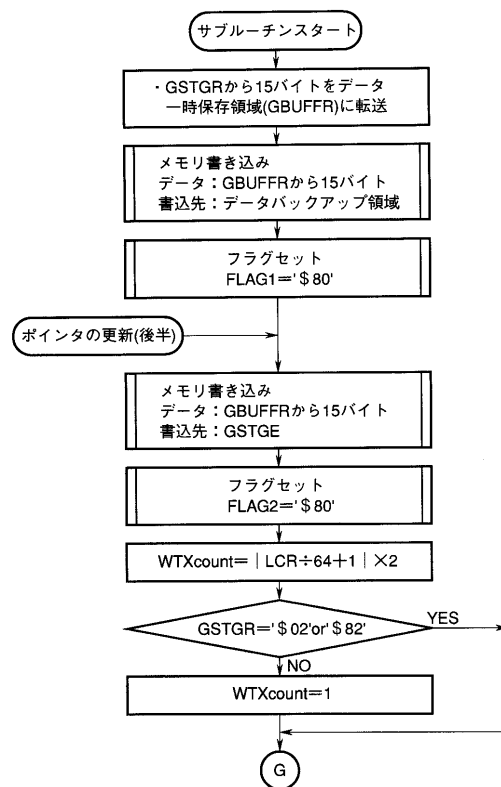
【図 31】



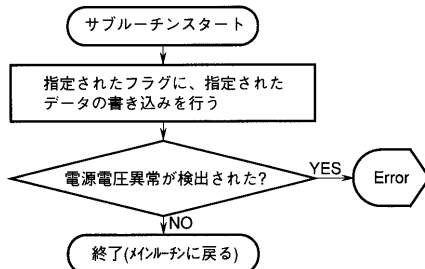
【図 32】



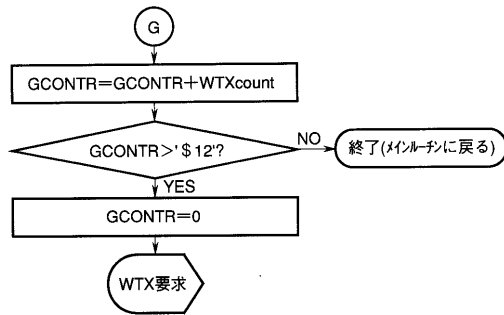
【図 34】



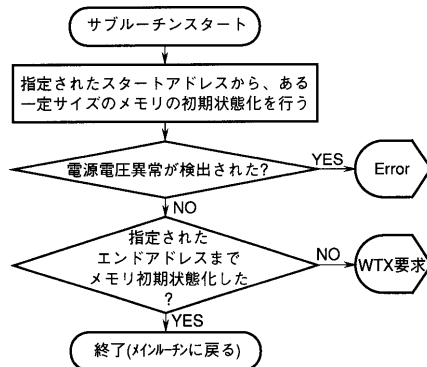
【図 33】



【図 35】



【図 36】



【図 38】

WAI	PFSN	(その他の情報)	DFSN	(DF名、その他の情報)
-----	------	----------	------	--------------

WAI	DFSN	(その他の情報)	FSIZ	USIZ	(その他の情報)
-----	------	----------	------	------	----------

【図 39】

WAI	DFSN	(その他の情報)	FSIZ	USIZ	(その他の情報)
-----	------	----------	------	------	----------

【図 40】

WAI	DFSN	(その他の情報)	EFTOP	EFSIZ	(その他の情報)
-----	------	----------	-------	-------	----------

【図 37】

コマンド処理中or終了

flag0	flag1	flag2	コマンドの実行ステータス
80	XX	XX	コマンド処理中(中断履歴)
00	XX	XX	コマンド正常終了

ステージ01,04,81,84

flag0	flag1	flag2	コマンドの実行ステータス
80	80	80	ポインタの本更新完了
80	00	80	フラグクリア
80	00	00	
80	80	00	ポインタのバックアップ完了

ステージ02,82,20,40

flag0	flag1	flag2	コマンドの実行ステータス
80	80	80	ポインタの本更新完了
80	00	80	データ領域のバックアップ完了
80	00	00	データ領域の本更新完了
80	80	00	ポインタのバックアップ完了

フロントページの続き

(72)発明者 中野 寛生

神奈川県川崎市幸区柳町 7 0 番地 株式会社東芝柳町工場内

審査官 前田 浩

(56)参考文献 特開平 0 9 - 2 3 1 1 1 0 (J P , A)

特開平 0 6 - 1 3 1 5 1 7 (J P , A)

特開平 0 7 - 0 4 4 4 2 6 (J P , A)

特開平 0 5 - 3 4 6 9 7 7 (J P , A)

(58)調査した分野(Int.Cl. , D B 名)

G06K 17/00-19/18