

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2010-271889

(P2010-271889A)

(43) 公開日 平成22年12月2日(2010.12.2)

(51) Int.Cl.	F 1	テーマコード (参考)
G06Q 20/00 (2006.01)	G06F 17/60 410E	
	G06F 17/60 410C	
	G06F 17/60 412	

審査請求 有 請求項の数 7 O L (全 36 頁)

(21) 出願番号	特願2009-122737 (P2009-122737)	(71) 出願人	000005223 富士通株式会社
(22) 出願日	平成21年5月21日 (2009. 5. 21)		神奈川県川崎市中原区上小田中4丁目1番1号
		(74) 代理人	100074099 弁理士 大菅 義之
		(74) 代理人	100133570 弁理士 ▲徳▼永 民雄
		(72) 発明者	青木 浩 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内
		(72) 発明者	細田 雅明 神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

最終頁に続く

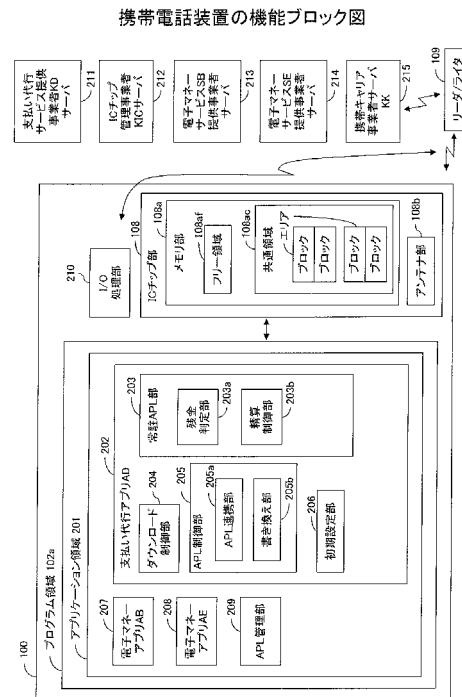
(54) 【発明の名称】 通信端末装置、方法、およびプログラム。

(57) 【要約】

【課題】電子マネーのチャージ残額や利用枠がゼロになった場合に、その電子マネーサービスに代わって支払い処理を行うアプリケーションを提供する。

【解決手段】携帯電話装置100は、支払い代行サービスを提供する支払い代行アプリケーションAD202を備える。支払い代行アプリケーションAD202は、支払い代行サービスを行うために、電子マネーサービスとの連携を行うAPL制御部205、代行金額を計算して支払い処理を行いまた支払い代行金額の精算を行う常駐APL部203、アプリケーションのダウンロードを行うダウンロード制御部204、アプリケーションの初期設定を行う初期設定部206、により支払い代行サービスを提供することを特徴とする。電子マネーサービスとの連携は、支払い代行アプリケーションAD202内に、電子マネーサービスの情報を書き込むことにより実現される。また、ICチップ部108のメモリブロックに書き込まれているエリアコードを書き換えることにより、支払い代行処理が可能となる。

【選択図】 図3



【特許請求の範囲】**【請求項 1】**

支払い代行アプリケーションにより提供される支払い代行サービスの属性種別情報を含む情報が格納されたメモリブロック内に、電子マネーアプリケーションにより提供される電子マネーサービスの属性種別情報を含む情報を書き込むことによって、前記支払い代行アプリケーションと、前記電子マネーアプリケーションを連携させる、APL連携部と、

電子マネーサービスの支払い可能範囲を越えた金額が要求されるエラー情報の発生元が、前記連携した電子マネーアプリケーションであるかを判定するAPL管理部と、

前記エラー情報の発生元が、前記連携した電子マネーアプリケーションであると判定した場合に、前記メモリブロック内から、前記連携した電子マネーアプリケーションにより提供される前記電子マネーサービスの情報を読み出し、前記メモリブロック内の前記支払い代行サービスの前記属性種別情報を、前記読み出した電子マネーサービスの前記属性種別情報に書き換える、属性種別情報書き換え部と、

前記書き換えられた前記支払い代行サービスの属性種別情報を用いることによって、前記連携した電子マネーアプリケーションに代わり、前記支払い可能範囲を超えた金額の支払い処理を実行し、前記支払い処理が正常に完了したことを示す情報を送信する残金制御部と、

を備える、通信端末装置。

【請求項 2】

前記連携した電子マネーアプリケーションへの電子マネーの入金処理を監視し、前記入金処理時において、前記支払い代行アプリケーションが代行して支払った代行金額がある場合に、前記代行金額を前記入金処理された金額から差し引いて精算し、前記精算した後の入金額を前記連携した電子マネーアプリケーションを管理するサーバに送信する精算制御部をさらに備えることを特徴とする、請求項 1 に記載の通信端末装置。

【請求項 3】

前記APL連携部は、一つの支払い代行アプリケーションと、一つ以上の電子マネーアプリケーションとを、それぞれ別個に連携させることを特徴とする、請求項 1 または 2 に記載の通信端末装置。

【請求項 4】

前記精算制御部は、前記通信端末装置の利用料金の精算処理を監視し、前記精算処理時に、前記支払い代行アプリケーションが代行して支払い処理した代行金額がある場合に、前記代行金額を、前記利用金額に追加することを特徴とする、請求項 2 に記載の通信端末装置。

【請求項 5】

前記支払い代行アプリケーションと前記電子マネーアプリケーションが連携したことを示す情報を含む解析キーを、暗号化して前記メモリブロック内に書き込むAPL制御部をさらに備えることを特徴とする、請求項 1 に記載の通信端末装置。

【請求項 6】

支払い代行アプリケーションにより提供される支払い代行サービスの属性種別情報を含む情報が格納されたメモリブロック内に、電子マネーアプリケーションにより提供される電子マネーサービスの属性種別情報を含む情報を書き込むことによって、前記支払い代行アプリケーションと、前記電子マネーアプリケーションを連携させるステップと、

電子マネーサービスの支払い可能範囲を越えた金額が要求されるエラー情報の発生元が、前記連携した電子マネーアプリケーションであるかを判定するステップと、

前記エラー情報の発生元が、前記連携した電子マネーアプリケーションであると判定した場合に、前記メモリブロック内から、前記連携した電子マネーアプリケーションにより提供される前記電子マネーサービスの情報を読み出し、前記メモリブロック内の前記支払い代行サービスの前記属性種別情報を、前記読み出した電子マネーサービスの前記属性種別情報に書き換えるステップと、

前記書き換えられた前記支払い代行サービスの属性種別情報を用いることによって、前

10

20

30

40

50

記連携した電子マネーアプリケーションに代わり、前記支払い可能範囲を超えた金額の支払い処理を実行し、前記支払い処理が正常に完了したことを示す情報を送信するステップと、
を備える、方法。

【請求項7】

支払い代行アプリケーションにより提供される支払い代行サービスの属性種別情報を含む情報が格納されたメモリブロック内に、電子マネーアプリケーションにより提供される電子マネーサービスの属性種別情報を含む情報を書き込むことによって、前記支払い代行アプリケーションと、前記電子マネーアプリケーションを連携させる処理と、

電子マネーサービスの支払い可能範囲を超えた金額が要求されるエラー情報の発生元が、前記連携した電子マネーアプリケーションであるかを判定する処理と、

前記エラー情報の発生元が、前記連携した電子マネーアプリケーションであると判定した場合に、前記メモリブロック内から、前記連携した電子マネーアプリケーションにより提供される前記電子マネーサービスの情報を読み出し、前記メモリブロック内の前記支払い代行サービスの属性種別情報を、前記読み出した電子マネーサービスの属性種別情報に書き換える処理と、

前記書き換えられた前記支払い代行サービスの属性種別情報を用いることによって、前記連携した電子マネーアプリケーションに代わり、前記支払い可能範囲を超えた金額の支払い処理を実行し、前記支払い処理が正常に完了したことを示す情報を送信する処理と、
をコンピュータに行わせることを特徴とするプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、電子マネー機能を提供する非接触型電子装置の技術に関する。

【背景技術】

【0002】

従来より、非接触型ICチップを用いたさまざまな電子マネーが利用されている。例えば、非接触型ICカード技術の利用方式であるFeliCa（登録商標）チップを用いた、Suica（登録商標）、PASMO（登録商標）、Edy（登録商標）等の前払い方式電子マネーや、QUICPay（登録商標）、iD（登録商標）等の後払い方式電子マネーがある。これらの電子マネーは非接触型ICチップを備えたICカードとしてだけでなく、携帯電話器内にFeliCaチップを搭載することによっても利用されている。例えば、「おサイフケータイ（登録商標）」は電子マネー機能搭載携帯電話器として知られている。

【0003】

前払い方式（プリペイド）の電子マネーでは、利用する前に現金を電子マネーに変換する「チャージ」を行い、チャージした電子マネーの金額がゼロになるまで支払いが継続される。また、ICチップをリーダ/ライタにかざした時に電子マネーの残高が予め設定した金額以下になっている場合、指定した金額が自動的にチャージされるオートチャージ方式も前払い方式の一つである。オートチャージ方式は、後日指定のクレジットカードで決済が行われる仕組みである。一方、後払い方式（ポストペイ）は、ICチップをリーダ/ライタにかざして料金を支払い、支払い後、利用額をクレジットカード決済する仕組みである。クレジットカード決済の場合には、予め指定した利用限度額に達するまで支払いが継続される。

【0004】

FeliCa対応携帯電話装置01内のFeliCaチップ02による電子マネーサービスは、図1に示されるように、フェリカネットワークス株式会社（以下、フェリカネットワークス）が運営、管理を行う、FeliCaチップ02の共通領域03を利用して提供されている。電子マネーサービスとそのサービスを提供するための電子マネーサービスアプリケーション（以下、電子マネーアプリ）を提供する電子マネーサービス事業者は、フェリカネットワークスに利用申し込みを行うことにより、電子マネーサービスを提供できる。フェリカネットワーク

スのリモート発行サーバ04の管理下で鍵05を用いることによって、FeliCaチップ02の共通領域03は、電子マネーサービス同士がそれぞれ独立してデータ管理ができるよう、疎の関係が構築されている。各電子マネーサービスの情報が疎の関係で管理されることで、各サービス事業者間の機密性が保たれているとも言える。

【0005】

また、フェリカネットワークスリモート発行サーバ04は、FeliCaチップ02内の共通領域03を利用する電子マネーサービスのデータが重複しないよう管理を行っている。例えば、共通領域03内のメモリ領域をエリアに分割して各サービス事業者に割り当てを行う。図1では、共通領域03は6つのエリアに分割され、エリアごとにサービスが割り当てられている。また、エリアには個別のエリアコードがつけられている。図1では、“0001”から“0005”までエリアコードがつけられている。また、フェリカネットワークスリモート発行サーバ04は個別エリアの鍵情報を払い出し、管理等を行っている。なお、フェリカネットワークスリモート発行サーバ04は、サービス毎に、その情報量に基づいて、1ブロックを16バイトとする個別エリアのブロック数(データ量)を決めることができる。

10

【先行技術文献】

【特許文献】

【0006】

【特許文献1】特開2001-216387号公報

【特許文献2】特開2006-139433号公報

【発明の概要】

20

【発明が解決しようとする課題】

【0007】

前払い方式では、FeliCaチップを搭載した端末を、FeliCaチップと通信して情報を読み書きするリーダ/ライタにかざして支払いをする際、支払い額が予めチャージしてある金額を越えた場合、エラーとなり、そのサービスでの支払いができない。すなわち、次にチャージするまで一時的に電子マネーサービスが停止してしまうことがある。例えば、Suicaで改札に入ろうとした場合、Suica残額が初乗り運賃未満であると、エラーとなって改札内に入れず、あらためてチャージしなければならない時などである。また、オートチャージの前払い方式や後払い方式では、支払い額もしくは利用額の積算額が利用限度額の上限を超えた場合、エラーとなり、その電子マネーサービスでの支払いができないことがある。したがって、サービスの一時的な停止を防ぐためには、それぞれの電子マネーサービス毎にチャージ金額の残額や、支払い上限までの残額を利用前に確認しておくことが必要となる場面があった。

30

【0008】

このように、ある電子マネーの残額がゼロになり一時的に電子マネーサービスが停止してしまった場合、ICチップ内に備えられている他の電子マネーに残額があったとしても、他の電子マネーアプリから電子マネーが補填されるような機能は備えていなかった。これは、電子マネーアプリ同士がそれぞれ関与しない疎の関係にあり、相互に連携していないことが一つの要因である。

【0009】

40

例えばSuicaとPASMOのように、FeliCaチップのフォーマット形式を同一にし、フェリカネットワークスにてエリアコードを相互利用するケースは存在している。しかしながら、フォーマット形式の異なる複数のサービスに亘って、連携するようなサービスは提供されていない。

【0010】

したがって、本発明は、各電子マネーのチャージ残額や利用枠がゼロになった場合に、その電子マネーアプリに代わって支払いを行うサービスの提供を可能にすることを課題とする。

【課題を解決するための手段】

【0011】

50

本明細書に記載の通信端末装置は、メモリ部に、APL連携部と、APL管理部と、属性種別情報書き換え部と、残金制御部とを備える。APL連携部は、支払い代行サービスの属性種別情報を含む情報が格納されたメモリブロック内に、電子マネーサービスの属性種別情報を含む情報を書き込むことによって、支払い代行アプリケーションと電子マネーアプリケーションを連携させる。APL管理部は、電子マネーサービスの支払い可能範囲を越えた金額が要求されるエラー情報の発生元が、連携した電子マネーアプリケーションであるかを判定する。属性種別情報書き換え部は、まず、エラー情報の発生元が、連携した電子マネーアプリケーションであると判定した場合に、メモリブロック内から、連携した電子マネーアプリケーションにより提供される電子マネーサービスの情報を読み出す。次に、メモリブロック内の前記支払い代行サービスの属性種別情報を、先ほど読み出した電子マネーサービスの属性種別情報に書き換える。残金制御部は、書き換えられた支払い代行サービスの属性種別情報を用いることによって、連携した電子マネーアプリケーションに代わり、支払い可能範囲を超えた金額の支払い処理を実行する。支払い処理が実行されると、その処理が正常に完了したことを示す情報をリーダー/ライターに送信する。

10

【0012】

この通信端末装置によれば、連携した電子マネーアプリケーションにより提供される電子マネーサービスの残額や利用枠がゼロになった場合に、支払い代行アプリケーションがその電子マネーサービスに代わって不足金額の支払い処理を実行することができる。したがって、電子マネーサービスを一時的に停止することなく、利用を継続することが可能となる。

20

【0013】

また、本明細書では、通信端末装置の各構成要素が提供する機能をコンピュータに行わせることで、当該コンピュータで当該通信端末装置を構築させるためのプログラムについても記載している。

【発明の効果】

【0014】

本明細書に記載の通信端末装置によれば、電子マネーサービスを一時的に停止することなく、利用を継続することができるという効果を奏する。

【図面の簡単な説明】

【0015】

30

【図1】FeliCaチップ共通領域を利用した電子マネーサービスの例である。

【図2】携帯電話装置のハードウェア構成図である。

【図3】携帯電話装置の機能ブロック図である。

【図4】電子マネーアプリの初期設定処理を示すフローチャートである。

【図5】ICチップ部のデータ構造およびアプリケーション設定の例である。

【図6】ICチップ部のデータ構造およびアプリケーション追加の例である。

【図7】支払い代行アプリケーションと電子マネーアプリの連携のイメージ図である。

【図8A】支払い代行アプリケーションを実施するための処理の流れを示した図である。

【図8B】支払い代行アプリケーションを実施するための処理の流れを示した図である。

【図8C】支払い代行アプリケーションを実施するための処理に伴う、ICチップ部内のデータ構造を示した図である。

40

【図8D】支払い代行アプリケーションを実施するための処理に伴う、ICチップ部内のデータ構造を示した図である。

【図8E】支払い代行アプリケーションの初期設定処理を示すフローチャートである。

【図9A】支払い代行アプリケーションを実施するための処理の流れを示した図である。

【図9B】支払い代行アプリケーションを実施するための処理の流れを示した図である。

【図9C】支払い代行アプリケーションを実施するための処理に伴う、ICチップ部内のデータ構造を示した図である。

【図9D】支払い代行アプリケーションを実施するための処理に伴う、ICチップ部内のデータ構造を示した図である。

50

【図9E】支払い代行アプリケーションを実施するための処理に伴う、ICチップ部内のデータ構造を示した図である。

【図10A】ICチップのアクセスパターンを示した図である。

【図10B】ICチップのアクセスパターンの利用例を示した図である。

【図10C】ICチップのアクセスパターンの利用例を示した図である。

【図11】支払い代行アプリケーションの精算方法の例を示した図である。

【図12A】支払い代行アプリケーションの精算方法の例を示した図である。

【図12B】支払い代行アプリケーションの精算方法の例を示した図である。

【発明を実施するための形態】

【0016】

以下、本発明の実施の形態を図面に基づいて説明する。

図2は、通信端末装置である携帯電話装置100のハードウェア構成の一例を示した図である。

【0017】

図2において、携帯電話装置100は、CPU101、ROM102、RAM103、カメラ部104、無線通信処理部105、アンテナ部105a、オーディオインタフェース部106、キー操作部107、ICチップ部108を備えている。

【0018】

CPU101は、この携帯電話装置100全体の動作を制御する中央演算処理装置である。ROM(Read Only Memory)102は、各種の制御動作をCPU101が行うための制御プログラムを格納する。ROM102には、プログラム領域102aとデータ領域102bが存在し、プログラム領域102a内には、CPU101が実行する制御プログラムが予め格納されている。また、プログラム領域102a内には、携帯電話装置100に後述する機能を提供するためにCPU101が実行するアプリケーションプログラムを格納する領域も存在する。

【0019】

RAM(Random Access Memory)103は、CPU101が制御プログラムを実行する際に必要に応じて使用する作業用の一時記憶領域を提供する。ROM102に格納されている制御プログラムをCPU101が読み出してその実行を開始すると、CPU101は後述する各種の制御処理を行って、後述する各種の機能を提供する。

【0020】

カメラ部104は、撮影対象となる被写体を撮影する処理を行う。カメラ部104は、図示しない、レンズ、受光部を含む。無線通信処理部105は、オーディオインタフェース部106で受信した音声データを変換した信号を変調した高周波信号を、アンテナ部105aに出力する。アンテナ部105aは、高周波信号を電波に変換して空間に放射すると共に、空間の電波を受信して高周波信号に変換する。オーディオインタフェース部106は、受信した音声データ等のデータを無線通信処理部105に転送するとともに、無線通信処理部105から転送されたデータを出力する。キー操作部107は、携帯電話装置100の表面に設けられ、ボタンスイッチ等の入力キーを利用者(ユーザ)が操作すると、その操作内容を示す操作データをCPU101に転送する。

【0021】

ICチップ部108は、無線通信可能な電子装置部であり、メモリ部108aとアンテナ部108bを備える。メモリ部108aは、電子財布機能に関するデータを格納する。アンテナ部108bは、ICチップ部内のデータを読み書きすることが可能な携帯電話装置外部のリーダ/ライタ109と通信する。

【0022】

この携帯電話装置100は、上述のハードウェアが実行する制御処理により実現される、通話機能、メール機能、カメラ機能、インターネット機能以外の付加機能として、電子財布管理機能を備えている。この電子財布管理機能は、電子マネーの入出金を管理する機能のほか、この実施例においては、支払い代行アプリケーションADと電子マネーアプリとの連携を可能にする機能も有している。

10

20

30

40

50

【 0 0 2 3 】

なお、この実施例ではICチップを用いた電子マネーを例として説明するが、他の無線通信可能な電子装置を用いて、電子マネー機能を提供してもよい。

さらに、この実施例では通信端末装置として携帯電話装置を例として説明するが、具体的な装置は携帯電話装置に限定されず、無線通信機能を搭載し、アプリケーションプログラムを自動的に起動することが可能な携帯式の端末であればよい。このような携帯式の端末としては、PDA (Personal Digital Assistants) やモバイル式のパーソナルコンピュータ、などがある。

【 0 0 2 4 】

次に、図3について説明する。図3は、図2の携帯電話装置100のROM102のプログラム領域102a内部の機能構成と、ICチップ部108の内部を示した図である。携帯電話装置100は、プログラム領域102a内部の各機能構成要素とICチップ部108により電子財布機能を提供する。この実施例においては、さらに、各種電子マネーアプリと連携して支払いをすることができる支払い代行サービスを提供する。

10

【 0 0 2 5 】

プログラム領域102a内部には、前述のように、アプリケーションプログラムを格納するためのアプリケーション領域201が存在する。アプリケーション領域201には、CPU101によって実行される電子マネー機能を提供するアプリケーションプログラムが格納されている。

【 0 0 2 6 】

次に、支払い代行アプリケーション(以下、支払い代行アプリ)AD202の機能を説明する。常駐アプリケーション部(以下、常駐APL部)203は、残金制御部203aと精算制御部203bを備える。残金制御部203aは、連携した電子マネーアプリの支払い可能な金額を越えた金額について、代行して支払い処理を行う。残金制御部203aは、さらに支払い処理が完了したことを示す情報の送信も行う。精算制御部203bは、電子マネーアプリへの入金処理を監視し、残金制御部203aが支払いを行った金額の精算処理を行う。精算制御部203bは、さらに、精算を行った後の入金額を、後述する電子マネーアプリのサーバに送信する処理も行う。ダウンロード制御部204は、支払い代行アプリAD202のダウンロードの実施および制御を行い、支払い代行アプリAD202のプログラムをアプリケーション領域201内に格納する。アプリケーション連携制御部(以下、APL制御部)205は、アプリケーション連携部(以下、APL連携部)205aと属性種別情報書き換え部(以下、書き換え部)205bを備える。APL連携部205aは、電子マネーアプリと支払い代行アプリAD202との連携を行う。書き換え部205bは、電子マネーアプリと支払い代行アプリAD202の連携に利用する属性種別情報を書き換える。APL制御部205は、支払い代行アプリAD202と電子マネーアプリが連携したことを示す情報を含む解析キーを、暗号化して後述するメモリブロックに書き込む。連携および書き換えの処理の詳細については後述する。初期設定部206は、支払い代行アプリAD202の初期化を行う。

20

30

【 0 0 2 7 】

アプリケーション領域201内には、支払い代行アプリAD202の他に、電子マネー機能を提供する電子マネーアプリAB207、電子マネーアプリAE208が格納される。また、ICチップ部108とリーダー/ライター109との通信を監視するアプリケーション管理プログラム(以下APL管理部)209も格納される。

40

【 0 0 2 8 】

これらのアプリケーションは、携帯電話装置100の初期設定時から既に格納されていてもよいし、初期設定後にネットワークを通じて、各アプリケーションによってサービスを提供する事業者のサーバ(211、213、214)からダウンロードされてもよい。ここで、211は支払い代行アプリAD202によって支払い代行サービスを提供する支払い代行サービス事業者サーバである。また、213は電子マネーアプリAB207によって電子マネーサービスSBを提供する電子マネーサービスSB提供事業者サーバである。同様に214は電子マネーアプリAE208によって電子マネーサービスSEを提供する電子マネーサービスSE提供事業者サーバで

50

ある。

【0029】

なお、上述したアプリケーションの機能は、プログラムを実行するCPU101によって提供される。

次に、ICチップ部108および、アプリケーションの実行に関連する構成について説明する。

【0030】

ICチップ部108のメモリ部108aは、フリー領域108afと共通領域108acに分割されている。さらに共通領域108acはエリアと呼ばれる領域に分割され、エリア内はブロックに分割されている。エリアには、サービスの属性種別情報を示すエリアコードが付与され、上述した電子財布機能を提供するサービス毎にエリアが割り当てられている。共通領域108acには、電子財布機能として、複数の電子マネーサービスや、支払い代行サービスのデータが格納される。また、例えば、電子マネーサービス以外の電子財布機能として、買い物を行った際にポイントを貯めるサービスや、電子マネーで映画や演劇等のチケットを購入するサービスのデータが格納されていてもよい。

10

【0031】

メモリ部108a内のデータは、リーダ/ライタ109によって、もしくは、アプリケーション領域201のアプリケーションの機能を制御するCPU101によって、読み書きされ得る。

アンテナ部108bは、リーダ/ライタ109と通信する。I/O処理部210は、携帯電話装置100の構成要素の一つであり、アンテナ部108bがリーダ/ライタ109と通信するデータを、携帯電話装置100が読み取れる形式に変換するインタフェース装置である。

20

【0032】

ICチップ管理事業者サーバ212は、ICチップ部108のデータ管理を行う事業者のサーバであり、ICチップ部108と、アプリケーション領域201内のアプリケーションを介して相互に通信する。また、携帯キャリア事業者サーバ215は、携帯電話装置100のサービスを提供する事業者のサーバであり、携帯電話装置100の通話やメール等のサービスを管理している。

【0033】

ここで、図4を参照して、電子マネーアプリの初期設定処理と、サービスが開始された場合の処理について、フローチャートで説明する。

30

まず、S401で、CPU101は、携帯電話装置100内に電子マネーアプリがインストールされているか確認する処理を行う。インストールされていない場合(Nの場合)、CPU101は、S402で、インターネットを介して電子マネーサービス提供事業者のサーバ(図3の213もしくは214)から電子マネーアプリをダウンロードする処理を行い、S403に進む。インストールされている場合(Yの場合)、S403に進む。次に、S404で、CPU101はアプリ活性に同意するかを、利用者に確認する処理を行う。利用者がアプリ活性に同意しなかった場合(Nの場合)、CPU101は、S405でアプリ活性処理を中断して終了する。S404で利用者がアプリ活性に同意した場合(Yの場合)、CPU101はアプリ活性処理を行い、アプリケーションを活性化する。

【0034】

40

続いて、S406で、CPU101は、電子マネーの支払い方式の選択処理を行う。前払い方式の場合、CPU101は特に処理を行わず、サービスが開始される。利用者は電子マネーをチャージすれば、電子マネー機能を利用できる。オートチャージの前払い方式および後払い方式の場合、CPU101は、電子マネーアプリに、クレジットカード決済を行うクレジットカードの情報を関連付ける処理を行う。このようにデータを関連付ける処理を以降「紐付け」と呼ぶ。なお、この処理の例としては、当業者によく知られている任意の方法が利用できる。電子マネーアプリとクレジットカードとの紐付け処理が完了すると、電子マネー機能が利用できる。精算処理においては、オートチャージの前払い方式ではオートチャージされた金額の総額が、後払い方式では利用金額の総額がクレジットカード事業者を介して請求される。精算処理は、従来用いられている任意の方法を用いて実行することができる。

50

【 0 0 3 5 】

次に図5および図6について説明する。図5および図6は、電子マネーアプリを例として、電子マネーサービスを開始する際の、アプリケーションの設定および追加の例と、それに伴うICチップ部108のメモリブロックのデータ構造を示している。

【 0 0 3 6 】

ここで、アプリケーションの設定手順と、アプリケーションの追加手順について説明する。以降説明に使用する電子マネーアプリAB207は、前払い方式（プリペイド）と後払い方式（ポストペイ）の両方の機能を備える電子マネーサービスを提供し、電子マネーアプリAE208は、後払い方式の電子マネーサービスを提供するものとする。

【 0 0 3 7 】

また、実施例としては記載しないが、設定および追加する電子マネーアプリの支払い方式は、オートチャージの前払い方式でもあってもよい。さらに、支払い方式の組み合わせは、異種であっても同種であっても可能である。また、この実施例では、二種の電子マネーアプリを設定しているが、後述する支払い代行サービスを提供するためには、一種以上の電子マネーアプリが設定されていればよい。なお、後述する処理により支払い代行アプリケーションが設定された後で、一種以上の電子マネーアプリを追加することも可能であるが、この実施例においては、先に電子マネーアプリが設定されている場合について説明する。

【 0 0 3 8 】

次に、図5を参照して、図4のフローチャートで説明した処理における、アプリケーションとICチップのデータ構造について説明する。図5では、図4の「サービス開始」までのステップが示されている。

【 0 0 3 9 】

まず、図5の「S11. アプリケーションとICチップを紐付け」で、CPU101は、ICチップ部108のメモリブロックBlock1およびBlock2の電子マネーサービスSBのデータと、アプリケーション領域201の電子マネーアプリAB207のデータとを関連付ける（紐付ける）処理を行う。

【 0 0 4 0 】

ここで、ICチップ108内のメモリブロックを2つ利用しているが、電子マネーサービスのデータ量に合わせて、使用するブロック数を変更することが可能である。また、電子マネーサービスSBのデータには、電子マネーサービス提供事業者KBがICチップ管理事業者KICと契約した際に、ICチップ部108の共通領域108acのエリアのうち、どのエリアの使用を契約したかを示すエリアコード（属性種別情報）が含まれている。また、エリアコードの他に、ICチップ部108内で使用するメモリブロックの番号、サービス内容を表す属性情報、メモリブロックのデータ保持形式、ならびに電子マネーの精算形式などの情報も含まれている。属性情報とは、そのアプリケーションが、電子マネーサービスなのか、ポイントサービスなのかといったサービスの種類を示す情報である。電子マネーの精算方式とは、電子マネーの精算が前払い方式の減算（チャージした金額から減算）か、あるいは後払い方式の加算（予め設定した金額までの加算）であるかといった、精算方式の種類を示す。

【 0 0 4 1 】

次に「S12. PIN設定（セキュリティ設定）」は、図4のS403、S404で処理が行われた際のデータ構造を示している。CPU101は、電子マネーサービスSBへのアクセス制限を示すPIN情報と、電子マネーサービスSBの属性情報を、ICチップ部108内に書き込む処理を行う。この処理を行うことにより、電子マネーアプリAB207の初期化が行われ、電子マネーサービスSBを開始する準備が整う。

【 0 0 4 2 】

次の「S13. サービス開始」は、図4のS406以降の処理に伴うデータ構造を示している。電子マネーサービスSBを利用して支払いが行われると、前払い方式の場合、S13-1に示すように、通信したリーダー/ライター109が、減算のパス値をメモリブロックに書き込む処理を行う。一方、後払い方式の場合、S13-2に示すように、リーダー/ライター109が、加算

10

20

30

40

50

のパーズ値をメモリブロック内に書き込む処理を行う。ここで、パーズ値は、0以上の整数である。

【0043】

リーダ/ライタ109により書き込み処理が行われると、一定量の処理結果は履歴情報としてブロック内に保存される。予め設定された規定数を上回る履歴情報は、電子マネーサービス提供事業者KBのデータベース（図3の213内）で管理される。以上説明した処理およびデータ構造により、電子マネー機能が利用可能となる。

【0044】

次に、図6は、電子マネーアプリAB207が既に設定されている携帯電話装置100に、新たに電子マネーアプリAE208を追加する場合の、アプリケーション領域201とICチップ部108のメモリ部108aのデータ構造を示している。図6のデータ構造は、図4を参照して説明した処理フローに伴うものである。

10

【0045】

まず、「S21.別アプリケーションの追加」で、CPU101は、図5のS11と同様に、ICチップ部108のメモリブロックBlock3の電子マネーサービスSEのデータと、アプリケーション領域201の電子マネーアプリAE208のデータとを紐付ける処理を行う。

【0046】

続いて、「S22.別アプリのPIN設定（セキュリティ設定）」で、CPU101は、図5のS12の処理と同様に、電子マネーサービスSEへのアクセス制限を示すPIN情報と、電子マネーサービスSEの属性情報を、ICチップ部108内に書き込む処理を行う。この処理を行うことにより、電子マネーアプリAE208の初期化が行われ、電子マネーサービスSEを開始する準備が整う。

20

【0047】

図6のS21およびS22の処理を行うことにより、電子マネーサービスSEが新たに設定される。この時、既に設定されている電子マネーサービスSBのデータと、新たに追加された電子マネーサービスSEのデータは、互いに独立してICチップ部108内に格納されており、ICチップ部108内のデータと、アプリケーション領域201内のアプリケーションのデータを紐付ける処理は、それぞれ独立して処理される。したがって、電子マネーサービスSBと電子マネーサービスSEの間では、情報の連携のない状態（疎の関係）が保持されることになる。このような疎の関係は、情報保護の観点では重要な要素である。

30

【0048】

図5および図6を参照して説明したように、アプリケーション領域201内には、一つ以上の電子マネーアプリを格納することが可能である。また、電子マネーアプリAB207を設定した後に電子マネーアプリAE208を追加したように、一つ以上のアプリケーションを追加することも可能である。さらに、アプリケーション領域201に空き領域があれば、二種以上、すなわち、電子マネーアプリAB207および電子マネーアプリAE208とは別の電子マネーアプリを追加することも可能である。なお、その際も、上述のように、それぞれの電子マネーサービスは疎の関係で管理されることとなる。

【0049】

次に、図7について説明する。図7は、この実施例において実現される、支払い代行アプリAD202の情報と、電子マネーサービスSBおよび電子マネーサービスSEの情報との連携のイメージ図である。一つの支払い代行アプリAD202と、電子マネーサービスSBおよび電子マネーサービスSEとが、それぞれ別個に関連付けられることにより、連携する。この時、電子マネーサービスSBと電子マネーサービスSEの間では、連携前と同様に、疎の関係が保たれたままである。支払い代行アプリAD202は、このような連携によって、連携した電子マネーサービスそれぞれに対して、個別に支払いの代行を行うことが可能となる。

40

【0050】

次に、図8Aから図8Dを参照して、支払い代行アプリAD202を実施するための処理の詳細、および処理に伴うデータ構造について説明する。

図8Aおよび図8Bは、支払い代行アプリAD202を実行するための各要素に関連して、

50

支払い代行処理の流れを示した図である。各要素とは、ICチップ部108、支払い代行サービス提供事業者KDサーバ（以下、支払い代行サービス提供事業者KD）211、電子マネーサービスSBサーバ213、電子マネーサービスSEサーバ214、携帯キャリア事業者KKサーバ（以下、携帯キャリア事業者KK）215、およびICチップ管理事業者KICサーバ（以下、ICチップ管理事業者KIC）212である。

図8Cおよび図8Dは、S803からS807の処理に伴う、アプリケーション領域201とICチップ部108のデータ構造を示した図である。

【0051】

図8Aおよび図8Bを参照して、処理の流れを説明する。まず、支払い代行サービス提供事業者KD 211は、サービスを提供するための事前準備として、S801で、ICチップ管理事業者KIC 212に、ICチップ共通領域108acの利用申請を行う。ICチップ管理事業者KIC 212は、ICチップ共通領域108ac内のメモリ領域（メモリブロック）、および属性種別情報であるエリアコードを、支払い代行サービス提供事業者KD 211に対して割り当てる。

10

【0052】

次に、S802で、支払い代行サービス提供事業者KD 211は、携帯電話装置100のキャリア事業者KK 215に、支払い代行アプリAD202の利用申請を行い、承認を受ける。S801およびS802を行うことにより、支払い代行サービスを提供するための事前準備が整う。なお、図面には記載していないが、これらの処理はサーバを介してデータの授受が行われ、サーバ内データベースにデータが蓄積される。

【0053】

20

続いて、携帯電話装置100のCPU101が、S803からS807で、支払い代行アプリAD202の初期設定処理（アプリ活性化）を行う。

なお、S803からS807の初期設定処理について図8Eにフローチャートが示されているが、詳細は後述する。

【0054】

初期設定処理は、まず、CPU101が、ネットワークを介して、支払い代行サービス事業者KD 211に支払い代行アプリAD202のダウンロードを要求する処理を行う。CPU101は、支払い代行アプリAD202をダウンロードし、アプリケーション領域201に格納する処理（インストール）を行う。S803は、支払い代行アプリAD202がブレインストールされていることが確認された場合（図8EのS821）には、行われなくともよい。

30

【0055】

次に、CPU101が、支払い代行アプリAD202の初期設定を行う（図8EのS823）。CPU101は、まず、S804で、ICチップ108のメモリブロックを支払い代行アプリAD202のデータ保持形式に合わせ、情報を書き込める状態に設定するフォーマットを行う。次に、CPU101は、S805で、支払い代行サービス事業者KD 211に対してアプリ活性化要求を行う。続いてS806で、S805のアプリ活性化要求に応じて、支払い代行サービス事業者KD 211は、ICチップ管理事業者KIC 212にシリアル認証の確認処理を行い、ICチップ管理事業者KIC 212から認証完了応答がなされる。S806の処理により、支払い代行アプリAD202が活性化される（図8EのS824）。

【0056】

40

続いてS807で、CPU101は、ICチップ108のメモリブロック内に、支払い代行サービスSDへのアクセス制限を示すPIN情報と、支払い代行サービスSDの属性情報を、書き込む処理を行う。このとき、属性情報は、支払い代行サービスSDが電子マネーであるという属性で書き込みされる。CPU101は、ICチップ108からの完了応答を受けて書き込み処理が完了したことを確認し、支払い代行サービス事業者KD 211との間でのパスワード設定を完了することで、支払い代行アプリAD202の初期設定処理を終了する。

【0057】

なお、この実施例では、支払い代行アプリAD202の設定前に、電子マネーアプリAB207および電子マネーアプリAE208が既に設定されているが、支払い代行アプリAD202が設定された後で電子マネーアプリが設定されてもよい。さらに、以降説明する連携処理についても

50

、支払い代行アプリAD202と電子マネーアプリを設定する順序に関係なく実行可能である。

【 0 0 5 8 】

続いて、図 8 B のS808～S811で、CPU101は、支払い代行アプリAD202と電子マネーサービスSBを連携する処理を行う。

まず、S808で、CPU101は、支払い代行サービス提供事業者KD 211を介して、電子マネーサービスSBサーバ214にアプリ連携要求を行う。電子マネーサービスSBサーバ214は、要求を受けて、連携用情報をCPU101に返信する。連携用情報には、エリアコード（属性種別情報）、電子マネーサービスSBのデータが格納されているメモリブロックの番号、属性情報、データ保持形式、電子マネーサービスSBの精算形式等が含まれている。連携用情報が返信されると、支払い代行サービス提供事業者KD 211は、ICチップ内に書き込まれている電子マネーサービスSBの情報の読み出し要求を行う。

10

【 0 0 5 9 】

次に、S809で、CPU101は、ICチップ内に書き込まれている電子マネーサービスSBの情報を読み出す処理を行い、ICチップ内に格納されている情報が読み出される。続いて、S810で、CPU101は、S809で読み出した電子マネーサービスSBの情報を、支払い代行サービスSDのデータが書き込まれているブロック（Block3、Block4）内に、書き込む処理を行う。CPU101は、電子マネーサービスSBの情報が書き込まれたBlockの番号、Block内のフォーマット形式、エリアコード、電子マネーの精算形式等の情報である解析キーを、暗号化して書き込んでもよい。解析キーは、支払い代行アプリAD202と電子マネーアプリAB207が連携したことを示す情報も含む。なお、解析キーとして、Block4とBlock3も連携しているという情報を保持してもよい。その後、CPU101は、ICチップ108からの完了応答を受けて書き込み処理が完了したことを確認する。

20

【 0 0 6 0 】

次に、S811で、CPU101は、支払い代行アプリAD202と電子マネーサービスSBが連携したこと示す情報を、支払い代行アプリAD202の格納されている領域内に書き込む処理を行う。次にCPU101は、S812で、支払い代行サービス事業者KD 211を介して、電子マネーサービスSBサーバ214にアプリ連携完了要求を行い、完了応答を確認する。S808からS812の処理をもって、支払い代行アプリAD202と電子マネーアプリAB207の連携が完了する。

【 0 0 6 1 】

続いて図 8 C および図 8 D を参照して、図 8 A および図 8 B の処理に伴うデータ構造を説明する。図 8 C の、「S81 . アプリケーション ICチップを紐付け」および「S82 . PIN設定（セキュリティ設定）」は、図 8 A のS803からS807の処理に伴う、アプリケーション領域201とICチップ108のデータ構造を示している。

30

【 0 0 6 2 】

図 8 A のS803の処理が行われると、「S81 . アプリケーション ICチップを紐付け」に示すように、ICチップ部108のメモリブロックBlock3およびBlock4の支払い代行サービスSDのデータと、支払い代行アプリAD202のデータとが紐付けられる。

【 0 0 6 3 】

続いて、図 8 A のS804の処理によってフォーマットの行われた結果のデータ構造が「S82 . PIN設定（セキュリティ設定）」に示されている。ブロック内には、後述する解析キー、相互利用サービス属性種別情報、電子マネーの精算形式としてパース値が減算か加算かの情報、支払い代行値（総額）、電子マネー使用履歴、等が書き込まれる。なお、この実施例では2つのBlock3およびBlock4が用いられているが、情報の書き込みに使用するブロック数は、支払い代行サービスSDのデータ量に合わせて変更することが可能である。

40

【 0 0 6 4 】

次に、図 8 A のS807の処理を行うことにより、「S82 . PIN設定（セキュリティ設定）」では、支払い代行サービスSDへのアクセス制限を示すPIN情報と、支払い代行サービスSDの属性情報が、ICチップ108内に書き込まれることとなる。

【 0 0 6 5 】

50

以上のようなデータ構造が作成されることにより、支払い代行アプリAD202の初期化が行われ、支払い代行サービスSDを開始する準備が整う。

続いて、S808からS812の処理に伴う、アプリケーション領域201とICチップ内のデータ構造を説明する。

【0066】

図8Dの「S83. サービス連携 サービス開始へ(相互利用)」では、上述した解析キーおよび支払い代行サービスSDの属性種別情報が、Block3に書き込まれている。Block3と4にデータが分割されて格納されているが、支払い代行サービスSDの情報量によって、データは1つのブロックに格納されてもよく、また、2つ以上のブロックに格納されてもよい。

10

【0067】

Block4は、パース値(減算)、パース値(加算)および、支払い代行した電子マネーの総額(支払い代行値)が書き込めるようCPU101によりフォーマットされている。このようなフォーマットにより、支払い代行アプリAD202は、連携する電子マネーサービスの支払い方式が、前払い方式か後払い方式に関わらず連携でき、支払い代行サービスSDを提供することが可能となる。

【0068】

次に図8Eについて説明する。図8Eは、支払い代行アプリAD202の初期設定処理(図8AのS803からS807)を示したフローチャートである。

まずS821で、CPU101は、携帯電話装置100のアプリケーション領域201内に支払い代行アプリAD202がプレインストールされているか確認する処理を行う。プレインストールされている場合(Yの場合)は、そのまま次のステップS823に進む。プレインストールされていない場合(Nの場合)は、S822で、CPU101が支払い代行アプリAD202をダウンロードする処理を行う。

20

【0069】

続いて、CPU101は、S823で、支払い代行アプリAD202を実行する処理を行う。次に、S824で、CPU101は、支払い代行アプリAD202のサービス開始に同意するか、利用者に確認する処理を行う。なお、このS824の処理をアプリ活性化と呼ぶこととする。利用者がアプリ活性化に同意する場合(Yの場合)には、S825で、CPU101が支払い代行アプリAD202のサービスを開始して(アプリ活性化して)、初期設定処理が終了する。利用者がアプリ活性化に同意しない場合(Nの場合)には、S826で、CPU101が初期設定処理を中断し、アプリ活性化を終了する。

30

【0070】

なお、図8Aから図8Eを用いて説明した連携処理を行った後であっても、図6のように新たな電子マネーアプリを追加設定し、追加設定した電子マネーアプリと支払い代行アプリとを連携させることも可能である。

【0071】

次に、図9Aから図9Eについて説明する。図9Aから図9Eは、支払い代行アプリAD202が実行され、電子マネーサービスSBに代わり支払い処理を行うための処理ステップおよび、処理に伴うデータ構造を示す。

40

【0072】

支払い代行サービスSDは、利用者が電子マネーサービスSBを利用して支払い処理を行おうとした際、支払い金額が電子マネーサービスSBのチャージ残高を超えてしまった場合に、実行される。

【0073】

一般的に、前払い方式の電子マネーサービスでは、支払いを要求される金額がチャージ残高を超えた場合、その電子マネーで支払いをすることができず、サービスは一時停止する。サービスが一時停止した場合には、再度チャージを行うことで、サービスが再開する仕組みとなっているものが多い。しかしながら、支払い代行アプリAD202を備える携帯電話装置100では、残額を越えた支払い要求があった場合にも、継続して支払い処理をする

50

ことが可能となる。

【 0 0 7 4 】

まず、電子マネーサービスSBでの支払いを行う場合、S901で、ICチップ部108が、リーダ/ライタ109と通信を行う。このとき、CPU101は、ICチップ部108とリーダ/ライタ109との通信による信号を監視する。

【 0 0 7 5 】

電子マネーサービスSBに、予めチャージされた電子マネーを超える金額が要求された場合、すなわち、電子マネーサービスSBサーバ214が支払い処理できる範囲（支払い可能範囲）を超えた場合、エラーが発生する。このエラーの情報は、ICチップ部108と通信したリーダ/ライタ109から送信される。CPU101は、リーダ/ライタ109から返信されたエラー情報を監視するとともに受信する。

10

【 0 0 7 6 】

次にS902からS904で、CPU101は、エラー情報の発生元が、連携した電子マネーサービスSBであるかを判定する処理を行う。具体的には、S902でリーダ/ライタ109から送信されるエラー情報の信号を読み込み、S903でエラー情報の内容から電子マネーサービスを特定する処理を行う。この実施例では、CPU101は、エラー情報の内容から電子マネーサービスSBでエラーが発生したことを特定する処理を行う。

【 0 0 7 7 】

続いてS904で、CPU101が、電子マネーサービスSBが支払い代行アプリAD202と連携したサービスであるかを判定する処理を行う。連携したサービスであると判定した場合には、S905で、CPU101が、図8BのS810で情報の書き込み処理を行ったICチップ108内に格納された電子マネーサービスSBの情報の読み出し要求を行う。ICチップ108が読み出し要求に回答すると、CPU101は、S810で暗号化して書き込んだ解析キーを用いて、電子マネーサービスSBに関する情報をアプリケーション領域201内に読み出す処理を行う。このとき、アプリケーション領域201には、電子マネーサービスSBのメモリブロックのフォーマット形式や、エリアコード（属性種別情報）などの情報が保持されている。

20

【 0 0 7 8 】

次に、S906で、CPU101は、ICチップ108内に書き込まれている支払い代行サービスSDのエリアコード（属性種別情報）を、S905で読み出した電子マネーサービスSBのエリアコードに書き換える処理を行う。エリアコードの書き換えは、例えば、支払い代行サービスSDのエリアコードを“1005”、電子マネーサービスSBのエリアコードを“0002”とすると、CPU101は、エリアコード“1005”を一時的にエリアコード“0002”に書き換える処理を行う。そして、CPU101は、S907で、ICチップ108に対してデータの書き換え要求を行い、ICチップ108の支払い代行サービスSDのデータが格納されているメモリブロック内（Block3）に、書き換えられたエリアコードを書き込む処理を行う。

30

【 0 0 7 9 】

S906およびS907の処理により、支払い代行サービスSDのエリアコードが電子マネーサービスSBのエリアコードと同じになると、リーダ/ライタ109は、支払い代行サービスSDを電子マネーサービスSBとして認識する。このようにエリアコードを、代行したい電子マネーサービスのエリアコードに書き換えることにより、支払い代行サービスSDは、電子マネーサービスに成り代わって支払い処理を行うことが可能となる。

40

【 0 0 8 0 】

続いて、支払い代行サービスSDが、電子マネーサービスSBに代わって行う支払い処理を説明する。まず、S908で、CPU101が、支払い代行アプリケーションAD202を起動し、電子マネーサービスSBに代わって支払い処理を行う。このとき、支払い代行サービス提供事業者KD 211は、ICチップ管理事業者KIC 212に対し、支払い代行サービスSDによる電子マネーサービスSBの継続利用要求を行う。ICチップ管理事業者KIC 212は、要求に応じて支払い代行サービスSDに対して認証処理を行い、支払い代行サービス提供事業者KD 211は、認証処理を受けて、支払い代行アプリAD202にOK応答を行う。

【 0 0 8 1 】

50

続いてS909で、CPU101は、S908で処理した支払い代行サービスの利用情報を、支払い代行サービスSDの情報が格納されているメモリブロック内に書き込む処理を行う。例えば、代行して支払った金額が160円である場合には、パース値（減算）として、ICチップのメモリブロック内に“160”が書き込まれ、かつ、支払い代行金額の総額が上書きされる。書き込み処理は、ICチップ部108からの完了応答によって完了する。また、ICチップ部108は、正常に支払い処理が完了したことを、リーダー/ライター109に送信する。

【0082】

上述の支払い代行処理により、連携した電子マネーサービスの残金が不足した場合にも支払い処理が行われることになるので、連携した電子マネーのサービスを継続することが可能となる。

【0083】

なお、利用者は、支払い代行アプリAD202により支払い代行可能な金額の上限を任意に設定することができる。CPU101は、利用者により任意に設定された代行可能な金額の上限を支払い代行アプリAD202内に記憶することが可能である。上限の設定を行うことにより、際限なく支払い処理を行ってしまうことを防ぐことができる。CPU101は、支払い代行金額の総額が、設定された金額の上限を超えた場合には、支払い処理を中止する制御を行う。なお、支払い代行可能な金額は、設定してもしなくともよい。

【0084】

続いて、CPU101は、S910で、S901からS909の処理を行うことによる支払い代行サービスSDの利用状況（利用実績情報）を、支払い代行サービス提供事業者KD 211に送信する。支払い代行サービス提供事業者KD 211は、携帯電話装置100の携帯キャリア事業者KK 215にメール送信要求を行うことにより、携帯キャリア事業者KK 215の通信網を介して、利用者に電子メールで利用実績情報を送信しても良い。なお、利用実績情報としては、代替したサービスの名称（この実施例では電子マネーサービスSB）、利用日時、利用金額等の利用した内容に関する情報が含まれる。なお、S910は、オプションの処理であり、実行されてもされなくともよい。

【0085】

次に、図9Cから図9Eを参照して、図9Aおよび図9BのS901からS910で実行された処理に伴う、アプリケーション領域201とICチップ内のデータ構造を説明する。まず、図9Cの「S91. 支払い代行サービス利用シーン」は、図9AのS901でエラー情報がリーダー/ライター109と通信するときのデータ構造である。ICチップ内の電子マネーサービスSBの情報が格納されているBlock1の、残金（支払い可能な金額）が“¥0”となっている。この状態をリーダー/ライター109が読み出し、どのブロックがエラーであるかという情報を含むエラー情報をICチップ部108に送信する。このとき、ICチップ部108内の支払い代行サービスSDのエリアコードは、Block3内に示されるように、“1005”、すなわち支払い代行サービスSDのエリアコードである。そして、支払い代行アプリAD202は休止状態である。

【0086】

次に図9Dの「S92.」は、図9AのS906の処理により、支払い代行サービスSDのエリアコード“1005”が、電子マネーサービスSBのエリアコード“0002”に書き換わった状態を示している。なお、書き換えが終わると、図9BのS908の処理により、支払い代行アプリAD202はアクティブな状態となる。一方で、電子マネーサービスSBを提供する電子マネーアプリAB207は、休止状態となる。

【0087】

続いて、図9Eの「S93.」は、支払い代行サービスSDの利用実績情報がICチップ部108のブロック内に書き込まれた状態を示している。電子マネーサービスSBは、前払い方式であるので、Block4の最上段のパース値（減算）の欄に、支払い代行値が書き込まれている。また、Block4の上から3段目の支払い代行値の総額の欄に、利用金額を加算した支払い代行金額の総額が上書きされている。さらに、Block4内（この実施例ではBlock4の上から4段目）に、支払い代行処理を行った日時や、利用金額、代行されたアプリケーションの名称が書き込まれている。

10

20

30

40

50

【0088】

ここで、図10A、図10Bおよび図10Cを参照して、ICチップのブロック内の情報を読み書きするためのアクセスパターンについて説明する。図10Aは、アクセスパターンの4つのタイプを示した図である。図10Bおよび図10Cは、アクセスパターンが各処理においてどのように利用されているかを示している。

【0089】

まず、図10Aを参照して、4つのアクセスパターンを説明する。1つめはランダムタイプであり、任意のブロックを読み書きする動作を行う。2つめはサイクリックタイプであり、古いデータに新しいデータを上書きする動作を行う。3つめはパースタイプであり、数値データとして、増減算や払い戻しの動作を行う。4つめはPINタイプであり、パスワードによりサービス情報へのアクセス制限の動作を行う。リーダ/ライタ109、および携帯電話装置100のCPU101は、これら4つのアクセスパターンを、実行する処理に合わせて利用している。

10

【0090】

次に、図10Bを参照して、図4を参照して前述した電子マネーアプリの設定処理における、アクセスパターンの利用例を説明する。「S101. アプリケーション ICチップを紐付け」の処理においては、ICチップ部108のブロック内に書き込まれた電子マネーサービスの情報は、携帯電話装置100のCPU101により読み書きされる。また、リーダ/ライタ109とICチップ部108が通信する場合には、リーダ/ライタ109によりブロック内の情報が読み書きされる。この時、携帯電話装置100のCPU101やリーダ/ライタ109は、ランダムタイプのアクセスパターンを利用して、指定したブロックにアクセスし、読み書きする処理を行う。

20

【0091】

「S102. PIN設定処理(セキュリティ設定)」においてPIN設定処理を行う場合には、PINタイプのアクセスパターンが利用される。PINタイプを利用することにより、パスワード制御によりアプリケーションへのアクセス制限を行うことができ、セキュリティ設定が可能となる。利用者は携帯電話装置100のキー操作部107を介して、パスワードを入力することができる。CPU101は、設定されたパスワードをPIN情報として、属性情報を管理する。

【0092】

続いて、図10Cを参照して、図8Dを参照して前述した支払い代行アプリAD202と電子マネーサービスSBの連携処理における、アクセスパターンの利用例を説明する。

30

まず、支払い代行サービスSDのエリアコードを、連携する電子マネーサービスSBのエリアコードに書き換える処理においては、サイクリックパターンが利用される。例えば、電子マネーサービスSBおよび電子マネーサービスSEが支払い代行サービスSDと連携する場合を例として説明する。図10C右側の「サイクリック例」の欄に示されるように、支払い代行サービスSDが、電子マネーサービスSBの支払い代行処理を行う場合には、ブロック内の支払い代行サービスSDのエリアコードが、電子マネーサービスSBのエリアコードに上書きされる。

【0093】

続いて、電子マネーサービスSEの支払い代行処理を行う場合には、ブロック内の電子マネーサービスSBのエリアコードが、電子マネーサービスSEのエリアコードに上書きされる。そして、再度電子マネーサービスSBの支払い代行処理を行う場合には、電子マネーサービスSEのエリアコードが、電子マネーサービスSBのエリアコードに上書きされる。サイクリックパターンを利用することにより、支払い代行サービスのエリアコードを、支払い代行したい電子マネーサービスのエリアコードに、随時書き換えることが可能となる。

40

【0094】

次に、支払い処理された支払い代行金額を書き込む際には、パースタイプが利用される。図10Cの右下のパース例の欄、「2. 電子マネーサービスSBで支払い代行利用」に示されるように、支払い代行アプリAD202が、電子マネーサービスSBに代わって支払い処理を行うと、支払い代行した料金150円が数値データとして加算される。次に、「3. 電子マ

50

「電子マネーサービスSEで支払い代行利用」に示されるように、支払い代行アプリAD202が、電子マネーサービスSEに代わって支払い処理を行うと、支払い代行した料金300円が数値データとして、加算される。そして、支払い代行金額の総額450円が数値データとして書き込まれる。ここで、数値データとして処理されるパース値は、正の整数であることに留意されたい。以上のように、ICチップ108内のデータは、4つのアクセスパターンを利用することにより、管理されている。

【0095】

続いて、図11、図12Aおよび図12Bを参照して、支払い代行サービスSDにより支払い代行を行った電子マネーを精算するための、精算処理について説明する。精算方法の一つとして、前払い方式の電子マネーサービスに電子マネーをチャージすることにより精算を行う方法がある。

10

【0096】

精算処理の詳細を図11を参照して説明する。まず、S1101で、利用者が電子マネーサービスSBに現金をチャージすることにより電子マネーが入金される。すなわち、ICチップ部108がリーダー/ライター109と通信し、チャージされた金額の数値データがICチップ部108内に書き込まれる。このようにチャージが行われるとき、ICチップ部108とリーダー/ライター109との通信（入金処理）を、CPU101が監視している。

【0097】

次に、S1102で、電子マネーサービスSBのサーバ213から支払い代行サービス提供事業者KDのサーバ211を介して、支払い代行アプリAD202に支払い代行処理した金額の有無の問い合わせが行われる。問い合わせに回答して、支払い代行アプリAD202が電子マネーアプリAB207に代わって支払い処理をした支払い代行金額があるかどうか、CPU101がICチップ内の情報を確認する処理を行う。

20

【0098】

次に、S1103で、CPU101は、ICチップ部108のメモリブロック内の情報を読み出す処理を行い、ICチップ部108は結果を応答する。ここで、結果とは、支払い代行サービスSDが電子マネーサービスSBに代わって支払った金額のデータである。CPU101は、ICチップ部108の応答結果のデータに基づいて電子マネーサービスSBの精算金額を確定する処理を行う。

【0099】

続いてS1104で、CPU101は、支払い代行アプリAD202が支払い代行処理した金額を精算する処理を行う。ここで、精算処理とは、ICチップ部108内の支払い代行金額のデータを書き換える処理である。すなわち、CPU101は、チャージ金額から、ICチップ部108内に書き込まれている電子マネーサービスSBの支払い代行金額の総額を差し引き（減算し）、精算金額を精算するとともに、電子マネーサービスSBの支払い代行金額の総額を書き換える処理を行う。

30

【0100】

例えば、支払い代行金額の総額が150円で、3000円がチャージされた場合、支払い代行金額の総額をチャージ金額が上回っているため、支払い代行金額の総額は“0”に書き換えられ、“3000”から“150”を差し引いた“2850”が残金となる。また、支払い代行金額の総額が3150円で、3000円がチャージされた場合、支払い代行金額の総額をチャージ金額が下回っているため、支払い代行金額の総額は、“3150”から“3000”を引いた“150”に書き換えられ、残金は“0”となる。

40

【0101】

ICチップ部108内へのデータの書き込み処理が完了すると、ICチップ部108は完了応答信号を送信する。CPU101は、ICチップ部108からの信号を受信し、支払い代行サービス提供事業者KD 211に支払い処理が完了したという情報を含む信号を送信する。

【0102】

続いて、S1105で、支払い代行サービス提供事業者KD 211では、S1104の完了応答を受けて、サーバ内の代行金額を精算し、さらにチャージされた金額から代行金額を引いた精算後の残額を計算する処理を行う。精算処理が行われると、電子マネーサービスSBには、チ

50

チャージした金額から支払い代行金額を差し引いた額の電子マネーがチャージされる。例えば、支払い代行金額の総額が150円で、3000円がチャージされた場合、3000円から150円を差し引いた2850円がチャージされる。

【0103】

なお、この一連の処理においてチャージされた電子マネーサービスSBと、チャージされていない他の電子マネーサービスのデータは互いに疎の関係で管理されている。さらに図7に示したように、支払い代行アプリAD202と各電子マネーアプリとの連携は、それぞれ独立している。したがって、S1101からS1105の処理が支払い代行アプリAD202と電子マネーアプリAB207の間で行われても、電子マネーアプリAE208などの他の電子マネーアプリのデータに関しては、変化しないことが理解されるであろう。

10

【0104】

精算処理の別の方法として、図12Aおよび図12Bに示すように、携帯電話装置100の利用料金の精算処理に支払い代行サービスの精算処理を上乗せして行う方法がある。

まず、図12AのS1201で、携帯キャリア事業者KK 215が、ネットワークを介して、携帯電話装置100の利用料金を確認するために、支払い代行サービス提供事業者KD 211に、料金の問い合わせを行う。このとき、S1201の問い合わせ処理に伴う信号をCPU101が監視している。次にS1202で、支払い代行サービス提供事業者KD 211は、支払い代行アプリAD202が代行して支払い処理した代行金額のうち、未精算の金額を確認する処理を行い、携帯キャリア事業者KK 215に未精算額を送信する。未精算額がなかった場合には、精算処理は終了する。

20

【0105】

S1203で、未清算額が携帯キャリア事業者KK 215に返信されると、S1204で、携帯キャリア事業者KK 215は、携帯電話装置の利用料金に未精算金額を追加して、支払い代行サービス提供事業者KD 211および、支払い代行アプリAD202に精算額情報を送信することにより、未請求額を利用者に請求する。S1205で、請求を受けた利用者が支払いを行うと、S1207で、支払い完了通知が携帯キャリア事業者KK 215から支払い代行サービス提供事業者KD 211に通知される。

【0106】

S1206で通知される支払い完了通知は、ICチップ管理事業者KIC 212には通知されないため、利用者が携帯電話装置100に備えられているアプリケーションを利用して通知を行う必要がある。そこで、支払いが完了したことをICチップ管理事業者KIC 212に通知する方法を説明する。なお、この実施例では、ワンタイムパスワードを利用した方法を例として説明するが、その他の方法であってもよい。

30

【0107】

まず、S1207で、支払い代行サービス提供事業者KD 211は、ICチップ管理事業者KIC 212にワンタイムパスワード（ワンタイム解除キー）を要求し、応答、すなわち解除キーを受け取る。次に、S1208で、支払い代行サービス事業者KD 211は、携帯キャリア事業者KK 215に、ワンタイム解除キーメールの送付依頼を行うと、支払い代行アプリAD202を介して、ワンタイム解除キーメールが携帯電話装置100に送付される。

【0108】

次に、S1209で、利用者は、支払い代行アプリAD202を起動し、携帯電話装置100のキー操作部107を用いてワンタイム解除キーを入力する。ワンタイム解除キーが入力されると、S1210で、支払い代行サービス提供事業者KD 211は解除キーを検証し、利用者が正規かどうか確認する処理を行う。

40

【0109】

正規利用者からの解除キーであることが確認できると、支払い代行サービス提供事業者KD 211はICチップ管理事業者KIC 212に解除確認応答を送信する処理を行う。ICチップ管理事業者KIC 212は解除確認応答を受けて、支払い代行サービス提供事業者KD 211にOK応答を送信する処理を行う。そして、支払い代行サービス提供事業者KD 211は、支払い代行アプリAD202にOK応答を送信する処理を行う。

50

【0110】

続いて図12BのS1211で、CPU101は、ICチップ108に格納されている支払い代行金額を精算する処理を行う。すなわち、メモリブロック内に書き込まれた支払い代行金額を消去する処理を行う。S1211の精算処理が行われると、ICチップ108は支払い代行アプリAD202に処理完了の応答を送信し、CPU101は、支払い代行サービス提供事業者KD 211に完了応答を行う。以上説明したように、図12AのS1201から図12BのS1211の処理を行うことにより、支払い代行金額の精算が完了する。

【0111】

なお、図11の処理と同様に、電子マネーサービス同士、および支払い代行アプリAD202と電子マネーサービスとの連携はそれぞれ独立しているため、図12AのS1201から図12BのS1211の処理が行われる際、電子マネーアプリAE208に関しては、処理が行われない。

【0112】

図8Aから図12Bを参照して説明した処理を終えると、支払い代行アプリAD202を備える携帯電話装置100を用いた、支払い代行処理が終了する。支払い代行アプリAD202を備えた携帯電話装置100によって上述した支払い代行サービスSDが提供される。この支払い代行サービスSDにより、支払い代行アプリAD202と連携した任意の電子マネーサービスの支払い処理を、残額不足などの支払いエラーによって中断することなく、継続することが可能となる。

【0113】

なお、以上までに説明した全ての実施形態に関し、さらに以下の付記を開示する。

(付記1)

支払い代行アプリケーションにより提供される支払い代行サービスの属性種別情報を含む情報が格納されたメモリブロック内に、電子マネーアプリケーションにより提供される電子マネーサービスの属性種別情報を含む情報を書き込むことによって、前記支払い代行アプリケーションと、前記電子マネーアプリケーションを連携させる、APL連携部と、

電子マネーサービスの支払い可能範囲を越えた金額が要求されるエラー情報の発生元が、前記連携した電子マネーアプリケーションであるかを判定するAPL管理部と、

前記エラー情報の発生元が、前記連携した電子マネーアプリケーションであると判定した場合に、前記メモリブロック内から、前記連携した電子マネーアプリケーションにより提供される前記電子マネーサービスの情報を読み出し、前記メモリブロック内の前記支払い代行サービスの前記属性種別情報を、前記読み出した電子マネーサービスの前記属性種別情報に書き換える、属性種別情報書き換え部と、

前記書き換えられた前記支払い代行サービスの属性種別情報を用いることによって、前記連携した電子マネーアプリケーションに代わり、前記支払い可能範囲を越えた金額の支払い処理を実行し、前記支払い処理が正常に完了したことを示す情報を送信する残金制御部と、

を備える、通信端末装置。

(付記2)

前記連携した電子マネーアプリケーションへの電子マネーの入金処理を監視し、前記入金処理時において、前記支払い代行アプリケーションが代行して支払った代行金額がある場合に、前記代行金額を前記入金処理された金額から差し引いて精算し、前記精算した後の入金額を前記連携した電子マネーアプリケーションを管理するサーバに送信する精算制御部をさらに備えることを特徴とする、付記1に記載の通信端末装置。

(付記3)

前記通信端末装置内の前記APL管理部は、前記支払い代行アプリケーションの情報を前記アプリケーション領域内に格納するダウンロード制御部をさらに備えることを特徴とする、付記1に記載の通信端末装置。

(付記4)

前記APL連携部は、一つの支払い代行アプリケーションと、一つ以上の電子マネーアプ

10

20

30

40

50

リケーションとを、それぞれ別個に連携させることを特徴とする、
付記 1 から 3 のうちいずれか 1 項に記載の通信端末装置。

(付記 5)

前記精算制御部は、前記通信端末装置の利用料金の精算処理を監視し、前記精算処理時に、前記支払い代行アプリケーションが代行して支払い処理した代行金額がある場合に、前記代行金額を、前記利用金額に追加することを特徴とする、付記 2 に記載の通信端末装置。

(付記 6)

前記 APL 管理部は、前記通信端末装置の無線通信可能な電子装置部と、該電子装置部内の情報を読み書き可能な外部リーダ/ライタとの通信を監視し、外部リーダ/ライタから送信される前記エラー情報を受信することを特徴とする、
付記 1 に記載の通信端末装置。

10

(付記 7)

前記支払い代行アプリケーションと前記電子マネーアプリケーションが連携したことを示す情報を含む解析キーを、暗号化して前記メモリーブロック内に書き込む APL 制御部をさらに備えることを特徴とする、付記 1 に記載の通信端末装置。

(付記 8)

支払い代行アプリケーションにより提供される支払い代行サービスの属性種別情報を含む情報が格納されたメモリーブロック内に、電子マネーアプリケーションにより提供される電子マネーサービスの属性種別情報を含む情報を書き込むことによって、前記支払い代行アプリケーションと、前記電子マネーアプリケーションを連携させるステップと、

20

電子マネーサービスの支払い可能範囲を越えた金額が要求されるエラー情報の発生元が、前記連携した電子マネーアプリケーションであるかを判定するステップと、

前記エラー情報の発生元が、前記連携した電子マネーアプリケーションであると判定した場合に、前記メモリーブロック内から、前記連携した電子マネーアプリケーションにより提供される前記電子マネーサービスの情報を読み出し、前記メモリーブロック内の前記支払い代行サービスの前記属性種別情報を、前記読み出した電子マネーサービスの前記属性種別情報に書き換えるステップと、

前記書き換えられた前記支払い代行サービスの属性種別情報を用いることによって、前記連携した電子マネーアプリケーションに代わり、前記支払い可能範囲を超えた金額の支払い処理を実行し、前記支払い処理が正常に完了したことを示す情報を送信するステップと、

30

を備える、方法。

(付記 9)

支払い代行アプリケーションにより提供される支払い代行サービスの属性種別情報を含む情報が格納されたメモリーブロック内に、電子マネーアプリケーションにより提供される電子マネーサービスの属性種別情報を含む情報を書き込むことによって、前記支払い代行アプリケーションと、前記電子マネーアプリケーションを連携させる処理と、

電子マネーサービスの支払い可能範囲を越えた金額が要求されるエラー情報の発生元が、前記連携した電子マネーアプリケーションであるかを判定する処理と、

40

前記エラー情報の発生元が、前記連携した電子マネーアプリケーションであると判定した場合に、前記メモリーブロック内から、前記連携した電子マネーアプリケーションにより提供される前記電子マネーサービスの情報を読み出し、前記メモリーブロック内の前記支払い代行サービスの前記属性種別情報を、前記読み出した電子マネーサービスの前記属性種別情報に書き換える処理と、

前記書き換えられた前記支払い代行サービスの属性種別情報を用いることによって、前記連携した電子マネーアプリケーションに代わり、前記支払い可能範囲を超えた金額の支払い処理を実行し、前記支払い処理が正常に完了したことを示す情報を送信する処理と、をコンピュータに行わせることを特徴とするプログラム。

【符号の説明】

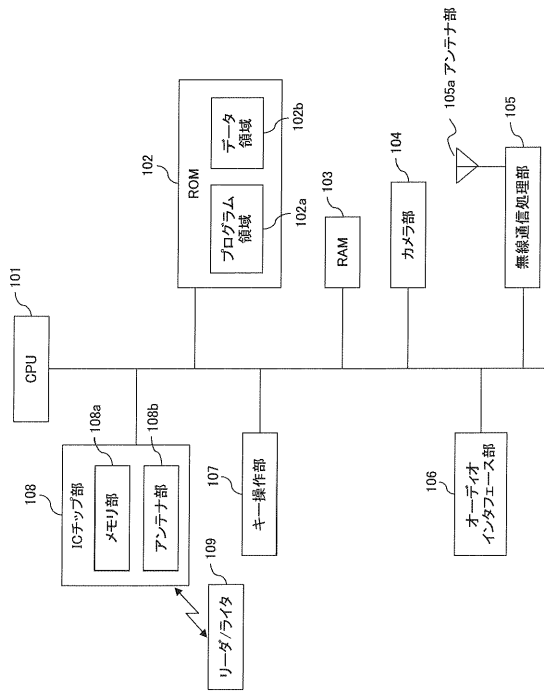
50

【 0 1 1 4 】

101	CPU	
102	ROM	
102a	プログラム領域	
102b	データ領域	
103	RAM	
104	カメラ部	
105	無線通信処理部	
105a	アンテナ部	
106	オーディオインタフェース部	10
107	キー操作部	
108	ICチップ部	
108a	メモリ部	
108af	フリー領域	
108ac	共通領域	
108b	アンテナ部	
109	リーダー/ライター	
201	アプリケーション領域	
202	支払い代行アプリケーションAD	
203	常駐アプリ部	20
203a	残金制御部	
203b	精算制御部	
204	ダウンロード制御部	
205	APL制御部	
205a	APL連携部	
205b	書き換え部	
206	初期設定部	
207	電子マネーアプリケーションAB	
208	電子マネーアプリケーションAE	
209	APL管理部	30
210	I/O処理部	
211	支払い代行サービス提供事業者KDサーバ	
212	ICチップ管理事業者KICサーバ	
213	電子マネーサービスSB提供事業者サーバ	
214	電子マネーサービスSE提供事業者サーバ	

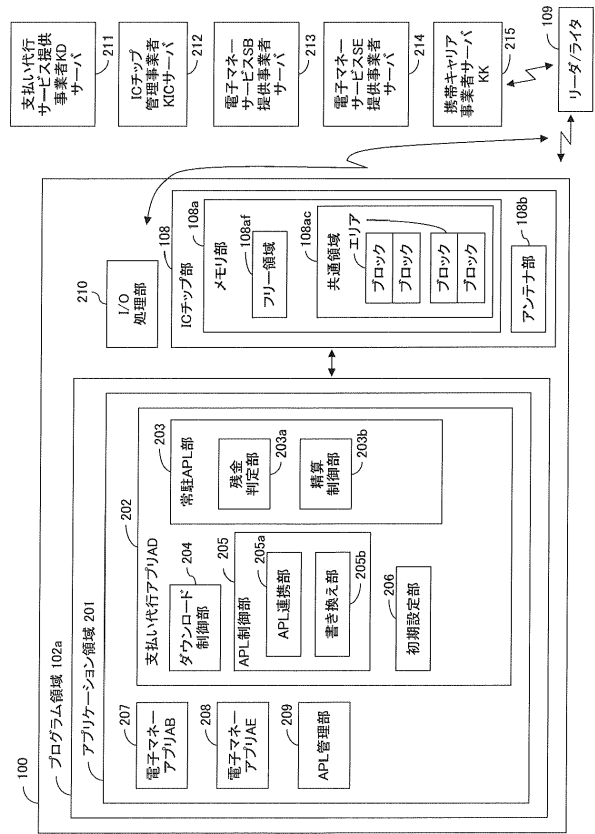
【図2】

携帯電話装置のハードウェア構成図



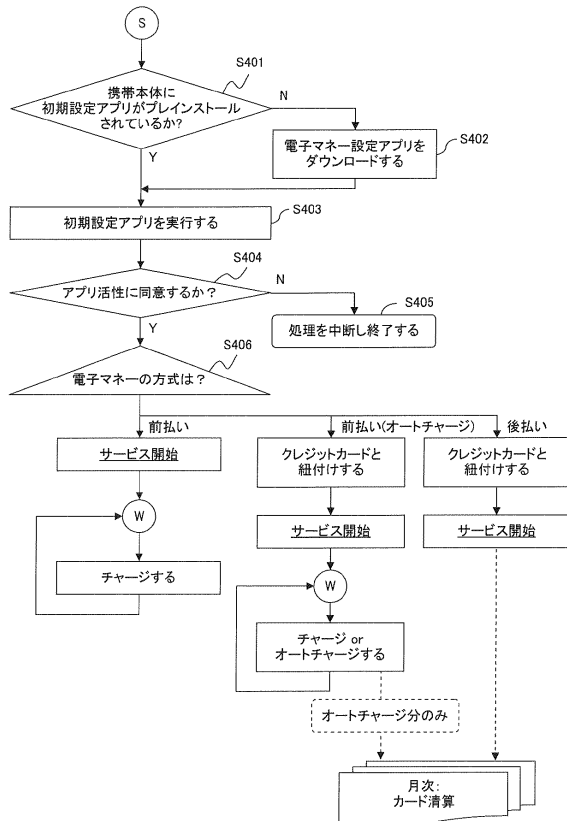
【図3】

携帯電話装置の機能ブロック図



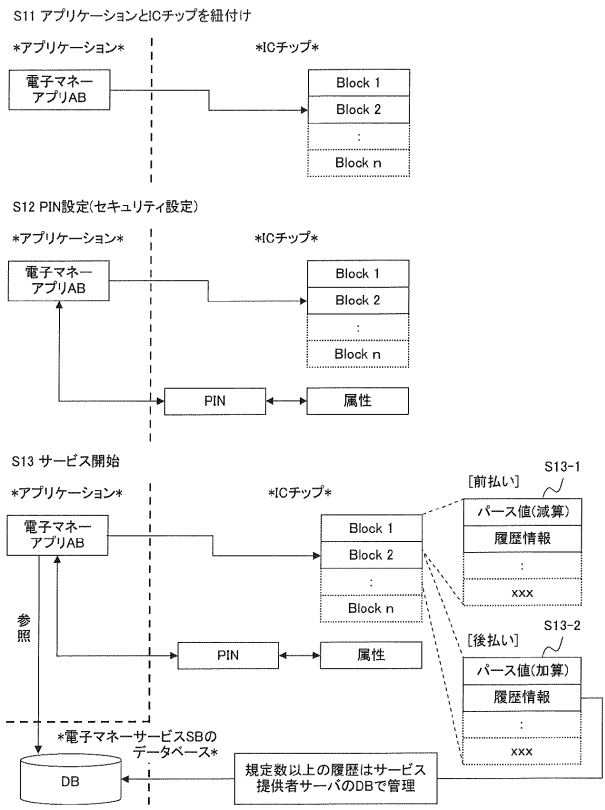
【図4】

電子マネーアプリの初期設定処理を示すフローチャート



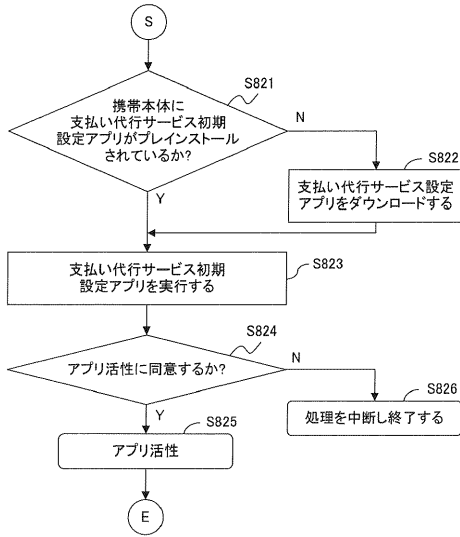
【図5】

ICチップ部のデータ構造およびアプリケーション設定の例



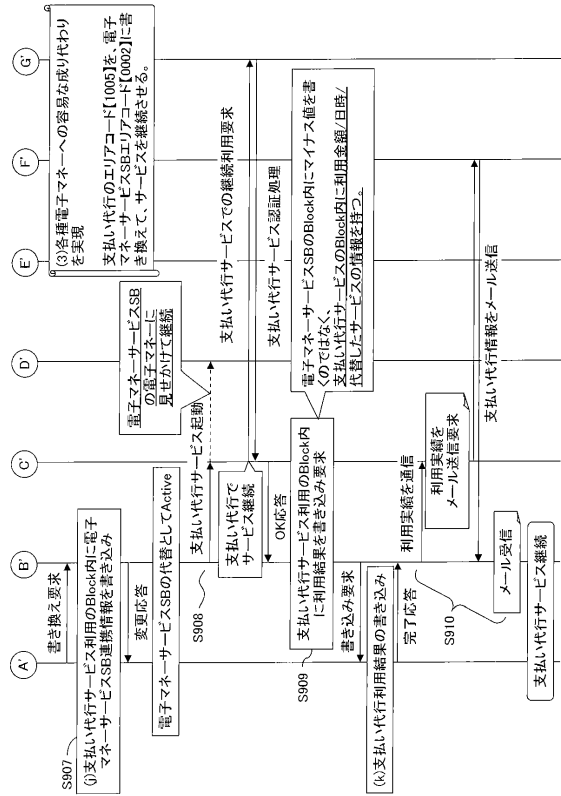
【 図 8 E 】

支払い代行アプリケーションの初期設定処理を示すフローチャート



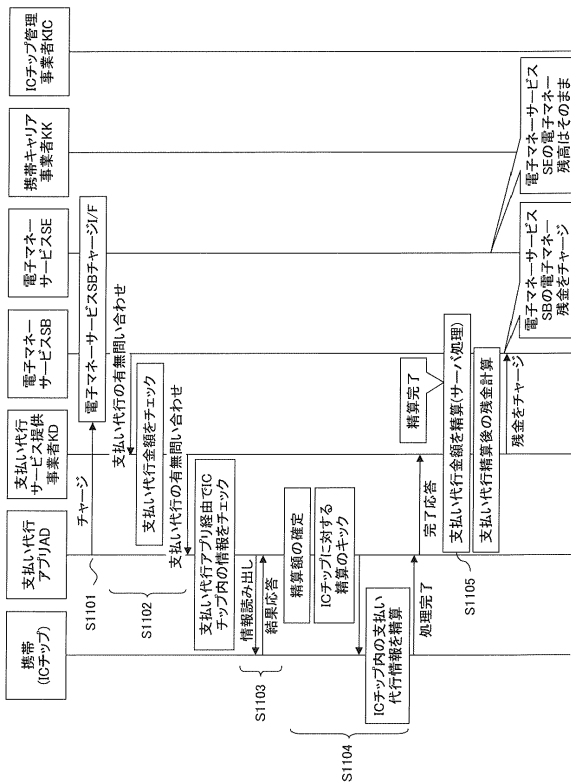
【 図 9 B 】

支払い代行アプリケーションを乗施するための処理の流れを示した図



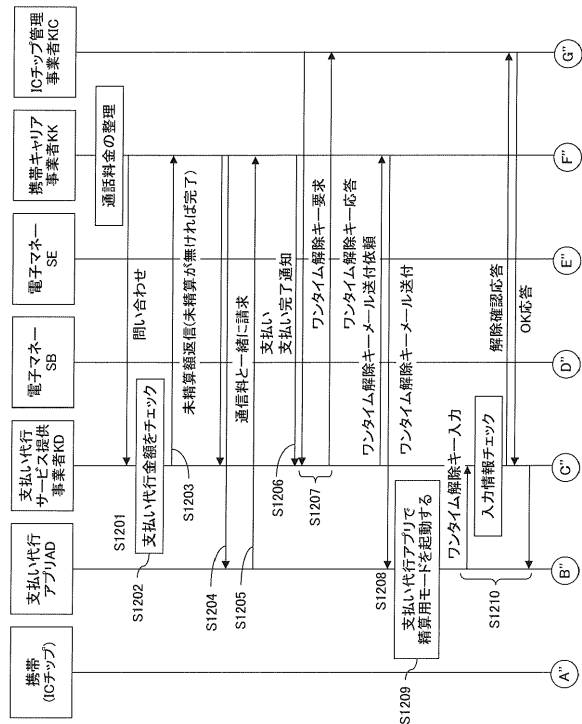
【 図 1 1 】

支払い代行アプリケーションの精算方法の例を示した図



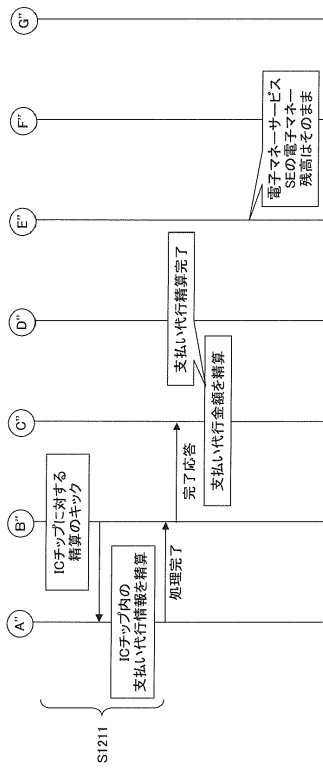
【 図 1 2 A 】

支払い代行アプリケーションの精算方法の例を示した図



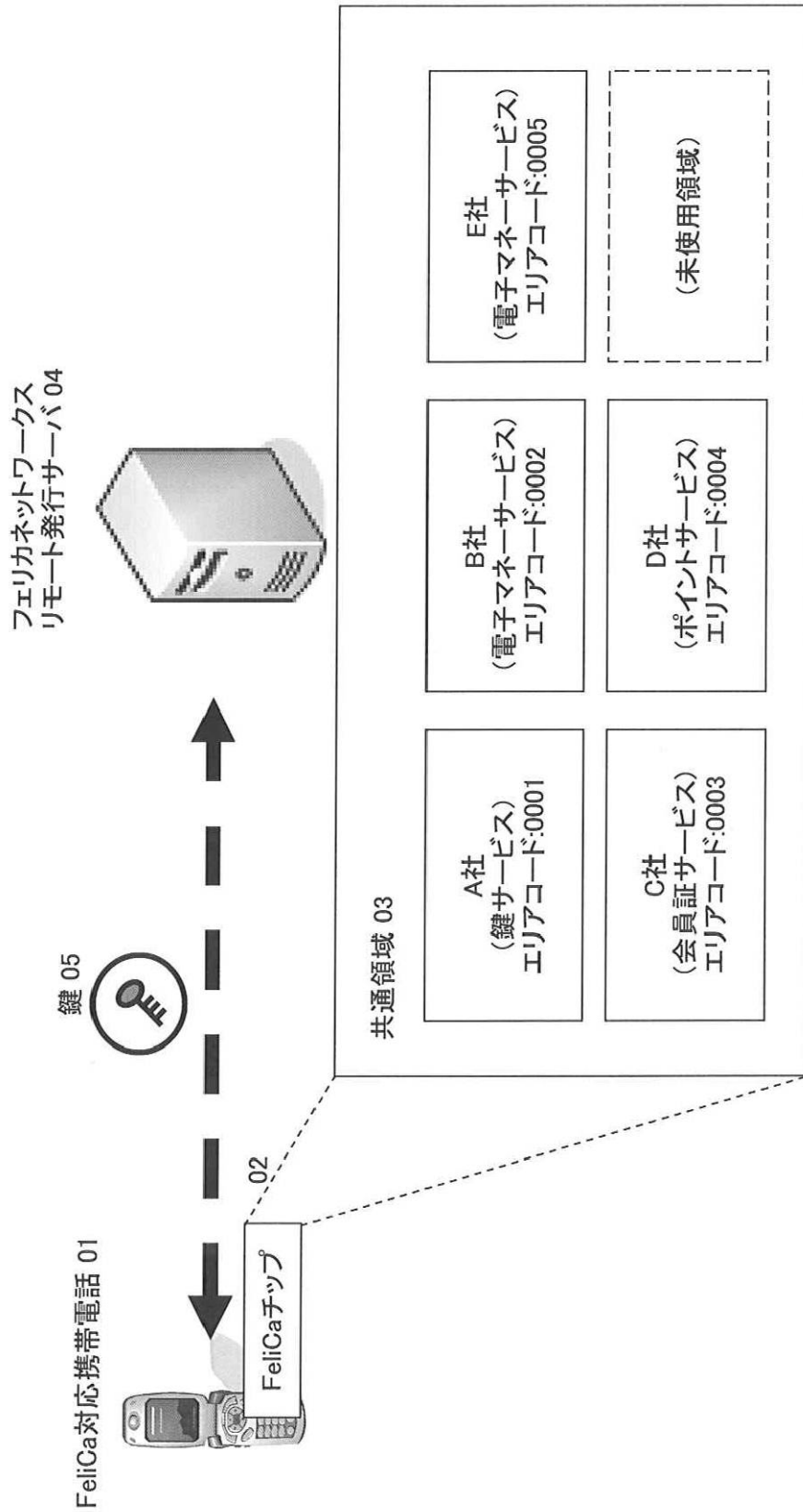
【 図 1 2 B 】

支払い代行アプリケーションの
精算方法の例を示した図



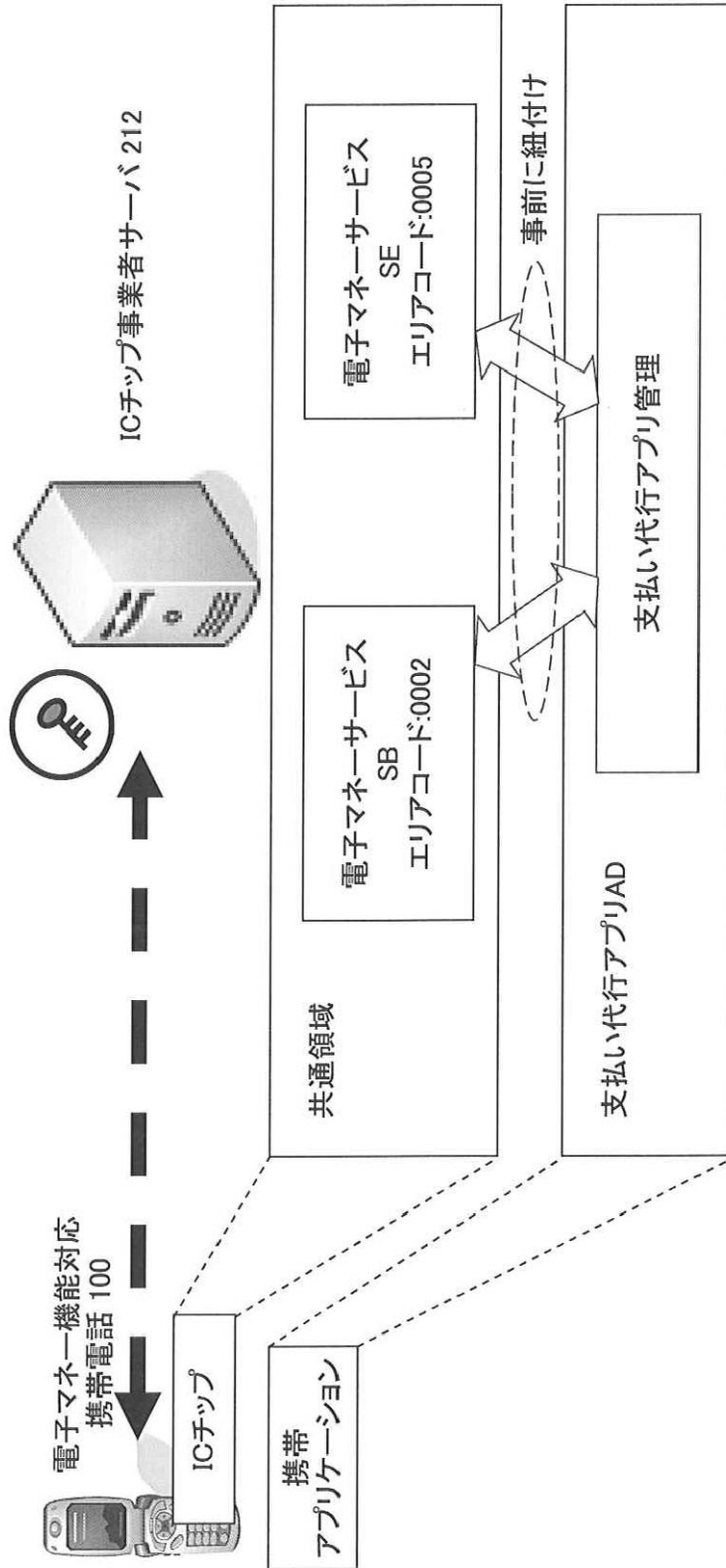
【 図 1 】

FeliCaチップ共通領域を利用した 電子マネーサービスの例



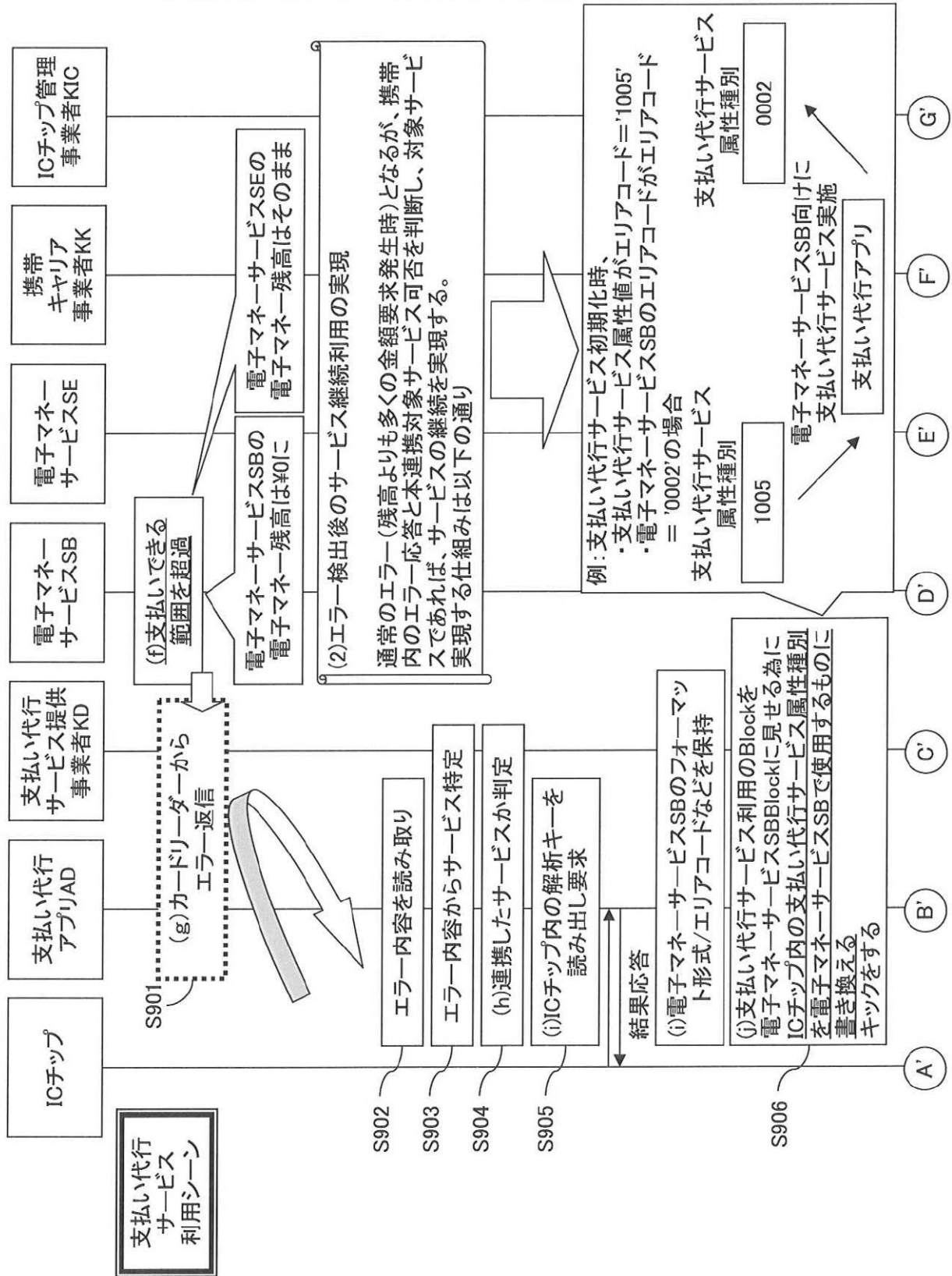
【図7】

支払い代行アプリケーションと 電子マネーアプリの連携のイメージ図



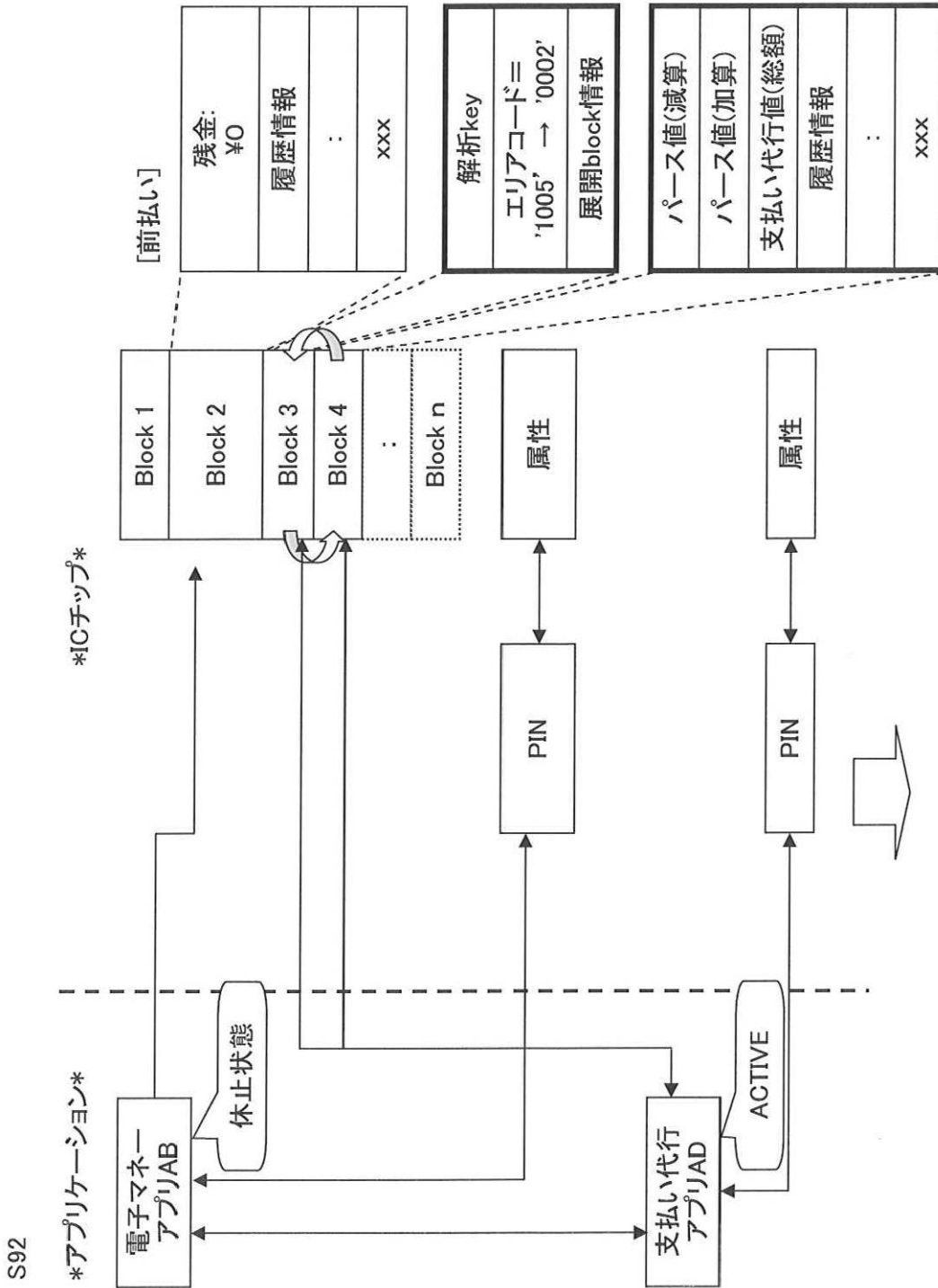
【図9A】

支払い代行アプリケーションを 実施するための処理の流れを示した図



【図9D】

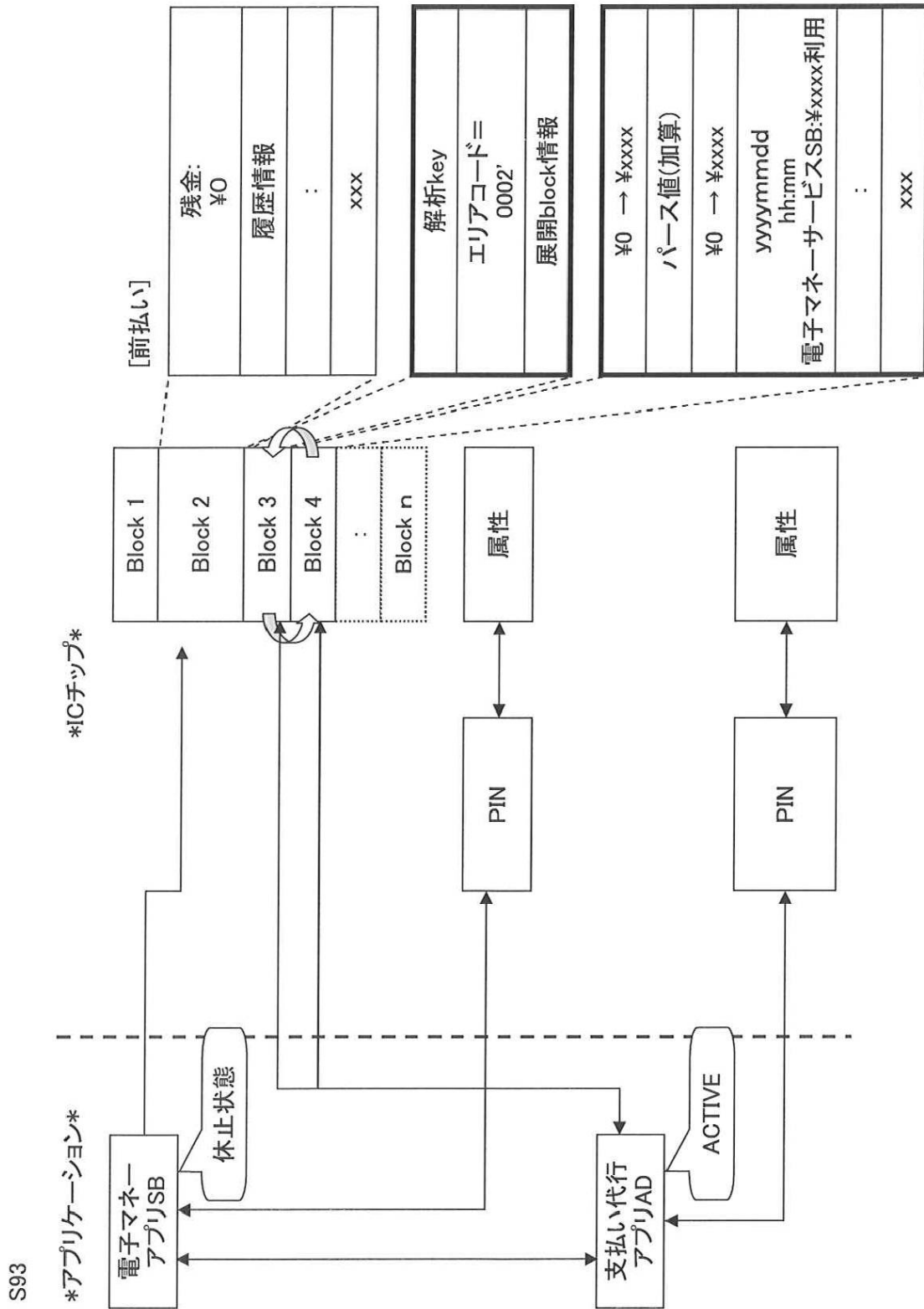
支払い代行アプリケーションを実施するための 処理に伴う、ICチップ部内のデータ構造を示した図



S92

【図 9 E】

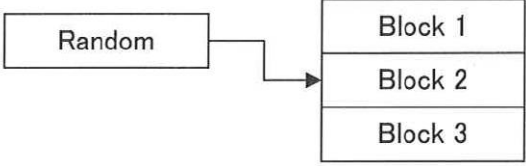
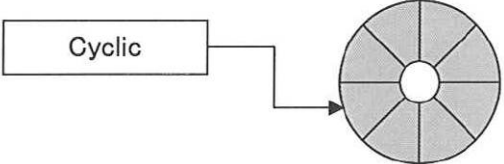
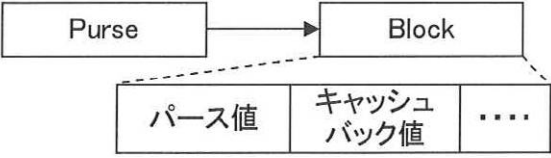
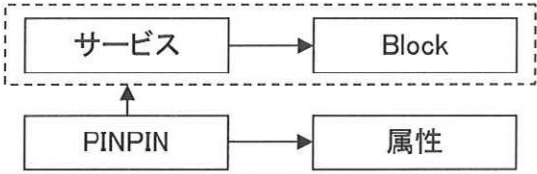
支払い代行アプリケーションを実施するための 処理に伴う、ICチップ部内のデータ構造を示した図



S93

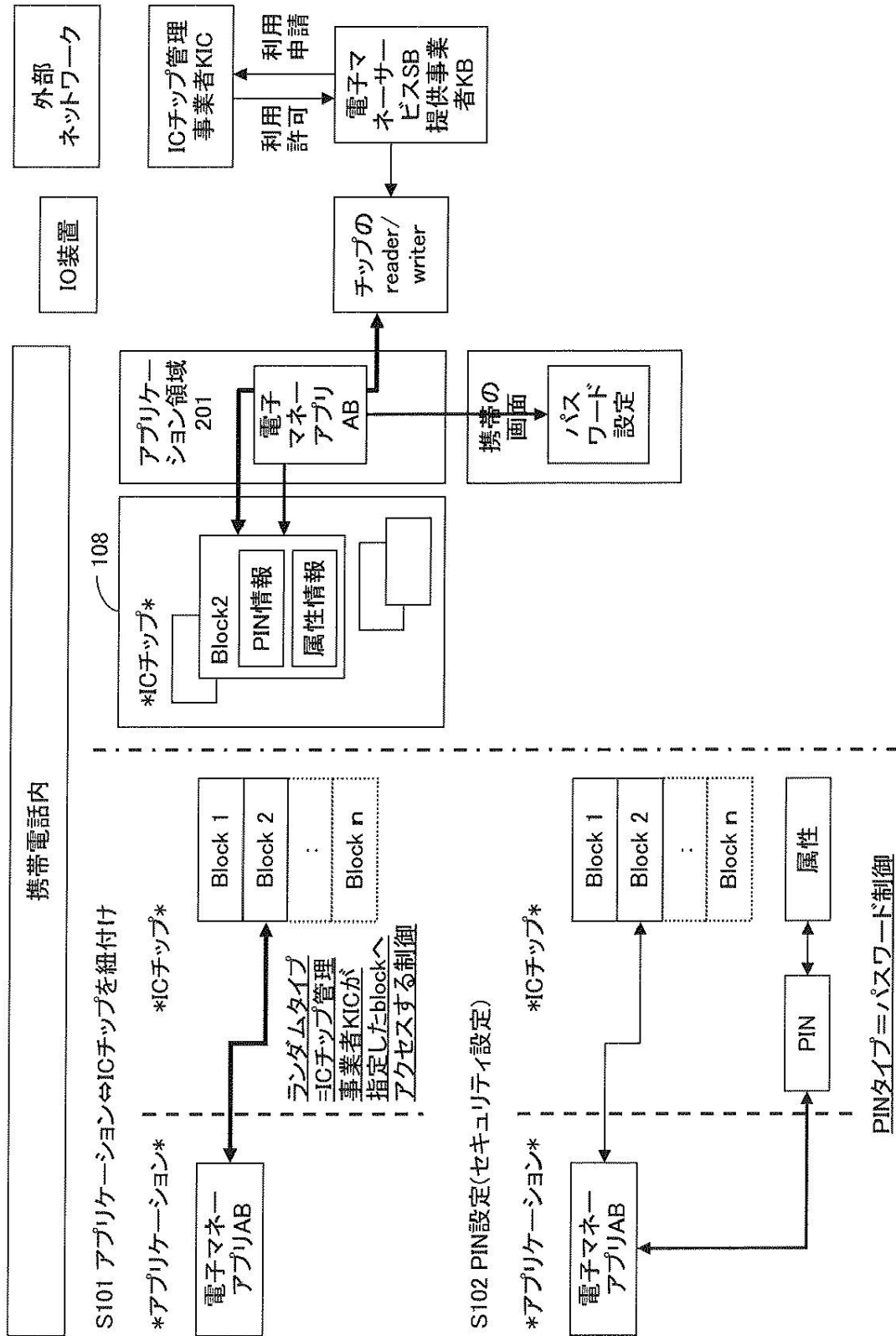
【図10A】

ICチップのアクセスパターンを示した図

タイプ	概要	イメージ図
ランダム	任意のブロックを読み書き	
サイクリック	古いデータを上書き	
パース	数値データとして増減算や払い戻し	
PIN	パスワードによるアクセス制限	

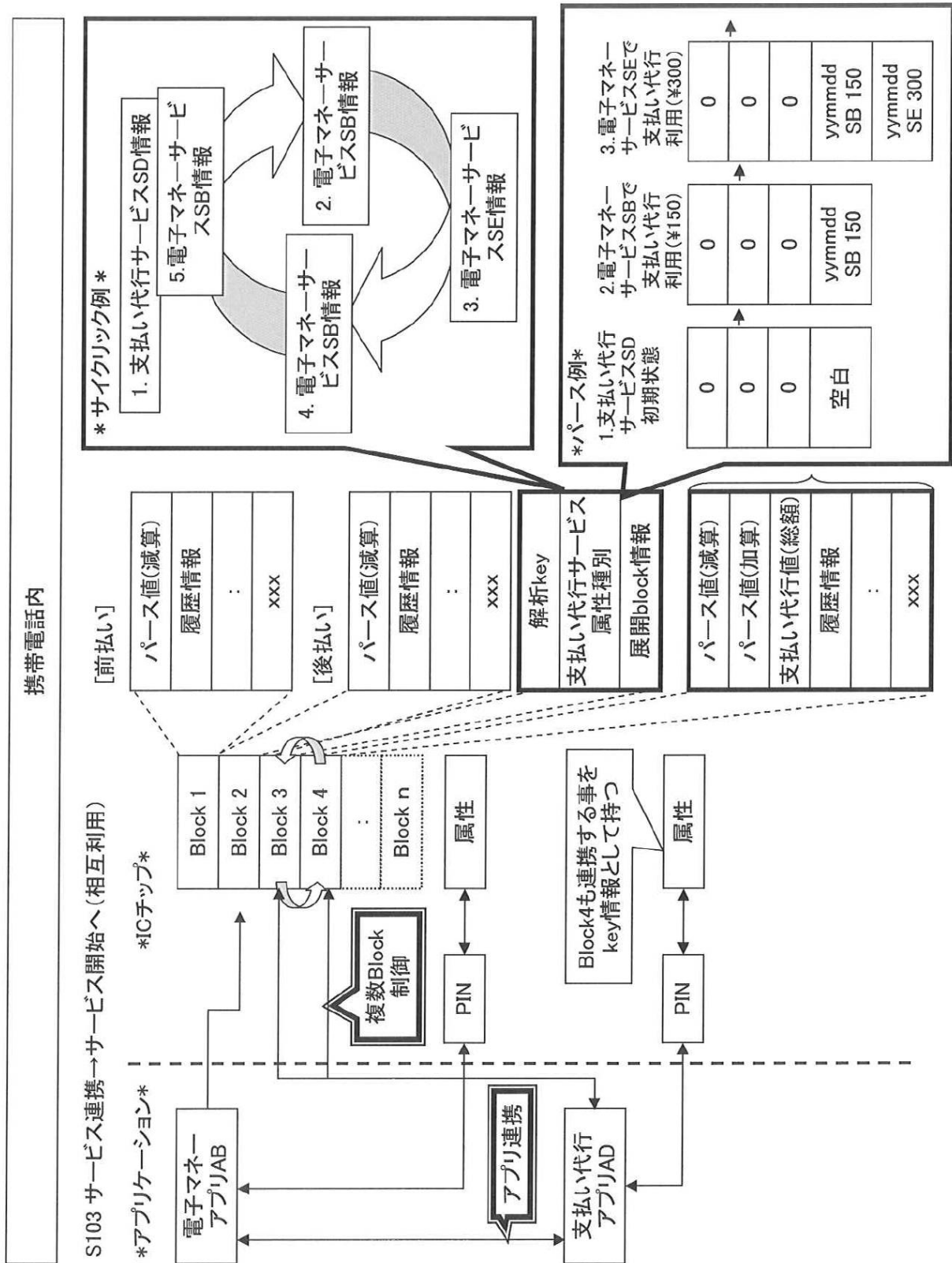
【図10B】

ICチップのアクセスパターンの利用例を示した図



【 図 1 0 C 】

ICチップのアクセスパターンの利用例を示した図



フロントページの続き

(72)発明者 増淵 康修

神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内

(72)発明者 森 敦子

神奈川県川崎市中原区上小田中4丁目1番1号 富士通株式会社内