

(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2007-58743

(P2007-58743A)

(43) 公開日 平成19年3月8日(2007.3.8)

(51) Int.C1.	F 1	テーマコード (参考)
<b>G06F 13/00</b> (2006.01)	G06F 13/00	354A 2C061
<b>G06F 1/32</b> (2006.01)	G06F 1/00	332Z 5B011
<b>G06F 3/12</b> (2006.01)	G06F 3/12	K 5B021
<b>B41J 29/38</b> (2006.01)	B41J 29/38	D 5B089
<b>HO4N 1/00</b> (2006.01)	B41J 29/38	Z 5C062
	審査請求 未請求 請求項の数 10 O L (全 27 頁)	最終頁に続く

(21) 出願番号	特願2005-245898 (P2005-245898)	(71) 出願人	000001007 キヤノン株式会社 東京都大田区下丸子3丁目30番2号
(22) 出願日	平成17年8月26日 (2005.8.26)	(74) 代理人	100071711 弁理士 小林 将高
		(72) 発明者	落合 将人 東京都大田区下丸子3丁目30番2号 キ ヤノン株式会社内
		F ターム (参考)	2C061 AP01 AP07 HH11 HT03 HT04 HT05 HT09 5B011 DA01 EB03 EB06 KK11 LL14 5B021 AA01 AA19 MM00 5B089 KB06 KG06 MB02 5C062 AA02 AA05 AA35 AB38 AB42 AB46 AB49 AC41 AC42 AC58 AF00 BA04

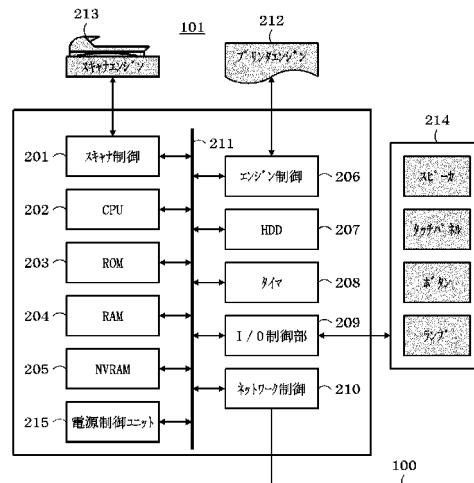
(54) 【発明の名称】ネットワークデバイスおよびネットワークデバイスの省電力制御方法

## (57) 【要約】

【課題】AppleTalkやTCP/IPのDHCPのようにアドレスが動的に変化する可能性のある環境下で動作するネットワークデバイスであっても、CPUへの電源を遮断した省電力状態に移行して省電力を実現できる仕組を提供すること。

【解決手段】CPU202は、省電力状態に移行する際、その時点で使用しているアドレスをRAM204に記憶させる。また、CPU202は、省電力状態が解除された際、RAM204に記憶されている前記アドレスが使用可能かどうか判断する。そして、該アドレスが使用可能な場合、該アドレスを使用するように制御する構成を特徴とする。

【選択図】図2



**【特許請求の範囲】****【請求項 1】**

所定の通信プロトコルに対応するアドレスを用いてネットワークを介した通信が可能であり、省電力状態へ移行可能なネットワークデバイスにおいて、

前記ネットワークデバイスが使用しているアドレスを記憶する記憶手段と、

前記省電力状態が解除された際、前記記憶手段に記憶されているアドレスが使用可能かどうか判断する判断手段と、

前記記憶手段に記憶されているアドレスが使用可能な場合、該アドレスを使用するよう

に制御する制御手段と、

を有することを特徴とするネットワークデバイス。

10

**【請求項 2】**

前記制御手段は、前記記憶手段に記憶されているアドレスが使用不可能な場合、新たなアドレスを使用するように制御する、ことを特徴とする請求項 1 記載のネットワークデバイス。

**【請求項 3】**

前記判断手段は、前記アドレスを含む情報をネットワークに出力し、その応答の有無に基づいて前記アドレスが使用可能かどうか判断する、ことを特徴とする請求項 1 又は 2 記載のネットワークデバイス。

**【請求項 4】**

前記ネットワークデバイスが前記省電力状態に移行する際、その時点で動作している処理を、前記省電力状態から復帰させるための制御情報を記憶する制御情報記憶手段と、

前記省電力状態が解除された際、前記制御情報記憶手段に記憶される前記制御情報を用いて、前記処理を前記省電力状態から復帰させる復帰手段と、

を有することを特徴とする請求項 1 ~ 3 のいずれかに記載のネットワークデバイス。

20

**【請求項 5】**

前記記憶手段は、前記ネットワークデバイスが使用している名前であって前記ネットワークにおいて当該ネットワークデバイスを識別するための名前を記憶し、

前記判断手段は、前記省電力状態が解除された際、前記記憶手段に記憶されている前記名前が使用可能かどうか判断し、

前記制御手段は、前記記憶手段に記憶されている名前が使用可能な場合、該名前を使用するように制御する、

ことを特徴とする請求項 1 ~ 4 のいずれかに記載のネットワークデバイス。

30

**【請求項 6】**

前記アドレスは、AppleTalkのプロトコルアドレスまたはTCP/IPのIPアドレスであることを特徴とする請求項 1 ~ 5 のいずれかに記載のネットワークデバイス。

40

**【請求項 7】**

前記記憶手段に記憶されているアドレスはネットワーク番号を含み、

前記判断手段は、前記記憶手段に記憶されているアドレスがルータのネットワーク番号の範囲内であるかどうかに基づいて前記アドレスが使用可能かどうか判断することを特徴とする請求項 1 ~ 5 のいずれかに記載のネットワークデバイス。

**【請求項 8】**

所定の通信プロトコルに対応するアドレスを用いてネットワークを介した通信が可能であり、省電力状態へ移行可能なネットワークデバイスの省電力制御方法において、

前記ネットワークデバイスが使用しているアドレスを記憶手段に記憶させる記憶ステップと、

前記ネットワークデバイスを省電力状態へ移行させる移行ステップと、

前記前記省電力状態が解除された際、前記記憶手段に記憶されているアドレスが使用可能かどうか判断する判断ステップと、

前記記憶手段に記憶されているアドレスが使用可能な場合、該アドレスを使用するよう

50

に制御する制御ステップと、

を有することを特徴とするネットワークデバイスの省電力制御方法。

【請求項 9】

請求項 8 に記載されたネットワークデバイスの省電力制御方法をコンピュータに実行させるためのプログラム。

【請求項 10】

請求項 8 に記載されたネットワークデバイスの省電力制御方法をコンピュータに実行させるためのプログラムをコンピュータが読み取り可能に記憶した記憶媒体。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、ネットワークデバイスの省電力制御に関するものである。

【背景技術】

【0002】

近年、より厳しくなってきている省エネルギー基準に対応するために、省電力モード時にCPUの電源を遮断する手段は既に実用化されている。

【0003】

しかし、その一方で、ネットワークに対応したプリンタや複合機等においては、省電力モードから高速に通常の動作モードに復帰する必要性が増している。また、最近の複合機等はプログラムの規模が増大し、ROMに全プログラムを格納するとコストアップが懸念されるため、ハードディスクを画像格納手段と兼用し、一部の領域にプログラムを格納していることがある。

【0004】

しかし、この場合は、起動後に、プログラムをハードディスクからRAMにダウンロードした後、RAM上でプログラムを実行することとなり、上記の高速復帰の必要性と相反する要素が存在する。

【0005】

そのため、RAM上に展開したプログラムを保持したまま、CPUの電源を遮断するような手段が提案されている（特許文献1）。

【特許文献1】特開2004-005029号公報

【発明の開示】

【発明が解決しようとする課題】

【0006】

このようにCPU電源を遮断するような省電力モードの実現形式において、デバイスがサポートするネットワークのプロトコルによってはつぎのような問題点がある。

【0007】

例えば、AppleTalk（登録商標）のプロトコルを例にとると、AppleTalkを使用しているネットワーク上のデバイスのそれぞれでは、そのプロトコルアドレスが動的に変化する。

【0008】

そのため、デバイスが省電力モードに移行している間に、そのデバイスが使用していたプロトコルアドレスが他のデバイスに割り当てられてしまう場合がある。即ち、省電力モード移行前に使用していたプロトコルアドレスは、他のデバイスと同一になってしまったため、使用することができなくなってしまう。

【0009】

これを解決するために、AppleTalkをサポートするデバイスが、使用しているプロトコルアドレスで問題がない（使用しつづけることが可能）かどうかを判断するべく、頻繁に省電力モードから復帰して、ネットワーク上のパケットを監視しようとする。しかしながら、このように頻繁に省電力モードから復帰していたのでは、実質、CPUの電源を遮断するような省電力モードには移行できないという問題点があった。

【0010】

次に、動的なプロトコルアドレスとしてTCP/IPのDHCPプロトコルを例にとる。このDHCP

10

20

30

40

50

プロトコルの場合、デバイスがDHCPサーバから取得したIPアドレスを定期的にサーバに更新要求する必要がある。そのため、そのIPアドレス更新間隔が短い場合、やはり、省電力モードに移行しにくいという問題点があった。

#### 【0011】

本発明は、上記の問題点を解決するためになされたものである。本発明の目的は、AppleTalkやTCP/IPのDHCPのようにアドレスが動的に変化する環境下で動作するネットワークデバイスであっても、省電力状態に移行して省電力を実現できる仕組を提供することである。

#### 【課題を解決するための手段】

#### 【0012】

本発明は、所定の通信プロトコルに対応するアドレスを用いてネットワークを介した通信が可能であり、省電力状態へ移行可能なネットワークデバイスにおいて、以下の構成を有することを特徴とする。ネットワークデバイスはネットワークデバイスが使用しているアドレスを記憶する記憶手段と判断手段と制御手段とを有する。また、前記判断手段は、省電力状態が解除された際、前記記憶手段に記憶されているアドレスが使用可能かどうか判断する構成である。さらに、制御手段は、前記記憶手段に記憶されているアドレスが使用可能な場合、該アドレスを使用するように制御する構成を有することを特徴とする。

#### 【発明の効果】

#### 【0013】

本発明によれば、アドレスが動的に変化する可能性のある環境下で動作するネットワークデバイスであっても、省電力状態に移行して省電力を実現できる等の効果を奏する。

#### 【発明を実施するための最良の形態】

#### 【0014】

##### 〔第1実施形態〕

以下、添付の図面を参照して本発明の実施例を詳細に説明する。

#### 【0015】

図1は、本発明の一実施形態を示すネットワークデバイスとして、ネットワーク接続機能を持つマルチファンクション装置がLANに接続されているシステムの概略図である。

#### 【0016】

図1において、101はLANに接続可能なマルチファンクション装置であり、本発明のネットワークデバイスに対応する。このマルチファンクション装置101は、AppleTalkプロトコルを用いたネットワーク印刷サービスやスキャンサービスをローカルエリアネットワーク(LAN)100に提供している。

#### 【0017】

また、同一のネットワーク上にLAN100を介して、AppleTalkプロトコルを搭載しているPC102が接続されている。また、PC103はAppleTalkのプロトコルルータである。

#### 【0018】

図2は、図1に示したマルチファンクション装置101の概略を表すブロック図であり、図1と同一のものには同一の符号を付してある。

#### 【0019】

図2において、201はスキャナエンジン制御部ユニットであり、スキャナエンジン213を制御する。202はCPUである。このCPU202は、マルチファンクション装置101全体の制御を行うための高機能で高速なCPUである。203はブートロム(ROM)であり、ブートプログラムが可能されている。204はRAM(メモリ)であり、CPU202のワーク領域として使用される。

#### 【0020】

205は不揮発性のRAM(NVRAM)であり、パネルから設定された値を保持しておくためのものである。206はエンジン制御部ユニットであり、プリンタエンジン212を制御する。207はハードディスクドライブ(HDD)であり、デバイスを制御するプログラム等が格納されている。またHDD207には、画像データや各種データ及びデバイス等を格納可能である。

#### 【0021】

10

20

30

40

50

208はタイマである。209はI/O制御部ユニットであり、スピーカ、タッチパネル、ボタン、ランプといったユーザインターフェース部分214を制御する。210はネットワーク制御ユニット210であり、LAN100との通信を制御する。215は電源制御ユニットである。この電源制御ユニット215を制御することによって、省電力モード（省電力状態）に移行することが可能となる。211はバスであり、上記201～210、215がそれぞれが接続されている。

#### 【0022】

電源が投入されると、CPU202は、ROM203からブートプログラムを読み出し、該ブートプログラムの制御に従ってデバイスのブートを実行する。また、CPU202は、ブートプログラムにしたがって、HDD207からデバイス制御プログラムをRAM204に展開し、該RAM204上に展開されたデバイス制御プログラムを読み出すことによって、デバイスの制御を実行することになる。10

#### 【0023】

図3は、図2に示した電源制御ユニット215によって、どのユニットの電源を遮断するかを示した図であり、図2と同一のものには同一の符号を付してある。

#### 【0024】

図3に示すように、電源制御ユニット215は、301で囲まれた部分（第1の電源系統）の電源をON/OFFすることが可能である。なお、301で囲まれた部分は、具体的にはスキャナエンジン213、スキャナ制御ユニット201、CPU202、ROM203、プリンタエンジン212、エンジン制御ユニット206、HDD207、タイマ208等を含む。なお、この301で囲まれた部分は、比較的消費電力が大きい。一方、301で囲まれた部分以外の部分（第2の電源系統）は、比較的消費電力が小さい。よって、この301で囲まれた部分への電源供給を遮断することにより、消費電力を低減することが可能となる。20

#### 【0025】

マルチファンクション装置101は省電力モードに移行する時、電源制御ユニット215が、301に示した範囲（第1の電源系統）の電源を遮断する。これによって、マルチファンクション装置101は省電力状態になって低電力が実現される。また、省電力状態を解除するべく省電力モードから通常の動作モードへ復帰する時には、この301に示した範囲に電源を投入することになる。なお、301で囲まれた部分以外の部分（第2の電源系統）は、省電力モード及び通常の動作モードの両モードにおいて電源供給が維持されている。

#### 【0026】

CPU202は、I/O制御209からの省電力モードへの移行指示やネットワーク制御ユニット210から一定時間ジョブが投入されない場合に、電源制御ユニット215に、301の範囲のユニットへの電源供給を遮断するように指示する。これにより、電源制御ユニット215は、301の範囲のユニットへの電源供給を遮断して、マルチファンクションデバイス101は省電力モードに移行することになる。30

#### 【0027】

また、電源制御ユニット215は、省電力モード時でも、I/O制御209やネットワーク制御ユニット210の各ユニットからPowerOn信号302を受信することが可能である。そして、電源制御ユニット215は、この信号を受信すると、301の範囲のユニットに電源を投入し、マルチファンクション装置101は省電力モードから通常の動作モードへ復帰することになる。40

#### 【0028】

図4は、図2に示したユーザインターフェース部分214の構成を説明する平面図である。

#### 【0029】

図4において、401は大型タッチパネルであり、ユーザはタッチパネルを操作することで、各種設定を行うことが可能である。図4の画面はコピーの待機画面に対応する。

#### 【0030】

402はテンキーボタンであり、「0」～「9」の数字を入力するのに使用する。「S」ボタンはサービスボタンであり、このボタンを押下することにより、タッチパネル上に各種サービス画面が出現し、コピー以外のサービスを行うことができる。「R」ボタンは設50

定ボタンであり、このボタンを押下することによって、タッチパネル上に、各種設定画面が出現し、パラメータの設定を行うことができる。

【0031】

403はスピーカであり、音声やブザー等をここから出力する。404はランプであり、印刷やコピーがジャムした場合には、ランプが点滅する。405は省電力へのON/OFFボタンであり、ユーザが省電力モードへの移行、および省電力モードから通常の動作モードへの復帰をボタン操作することが可能である。

【0032】

以下、図5～図8を参照して、本発明のネットワークデバイスにおける起動時のAppleTalkのプロトコル処理モジュール群の立ち上がりの処理について説明する。

10

【0033】

図5は、本発明のネットワークデバイスにおける第1の制御処理手順の一例を示すフローチャートであり、起動時のAppleTalkのプロトコル処理モジュール群（後述する図11の1101～1104）の立ち上がりの処理の流れに対応する。なお、このフローチャートの処理は、図2に示したCPU202が、HDD207に格納されたプログラムをRAM204上に読み出して実行することにより実現される。なお、図5中、S501～S506は各ステップを示す。

【0034】

電源が投入されると、CPU202は、各デバイスユニットの初期化処理を行う。そして、マルチファンクション装置101のAppleTalkのサービスが有効な設定にされていると、CPU202は、AppleTalkのサービスをネットワーク上に提供するために、本フローチャートに示す起動処理を実行する。以下、詳細に説明する。

20

【0035】

まず、ステップS501において、CPU202は、アドレス管理モジュール1103（図11）を起動する。そして、このアドレス管理モジュール1103は、ステップS502にて、プロトコルアドレスの解決（アドレス解決処理）を行う。なお、このアドレス解決処理の詳細は後述する図6、図7に示す。

【0036】

次に、アドレス管理モジュール1103は、ステップS503にて、名前の解決（名前解決処理）を行う。なお、この名前解決処理の詳細は後述する図8に示す。

30

【0037】

アドレス及び名前が決定した後、CPU202は、ステップS504において、プロトコルスタックモジュール1102（図11）を起動する。次に、CPU202は、ステップS505において、プリントサービスモジュール1101（図11）を起動する。この結果、ステップS506において、CPU202は、AppleTalkの印刷サービスを開始することが可能となる。

【0038】

図6、図7は、本発明のネットワークデバイスにおける第2の制御処理手順の一例を示すフローチャートであり、図5のステップS502に示したアドレス解決処理に対応する。なお、このフローチャートの処理は、図2に示したCPU202が、HDD207に格納されたプログラムをRAM204上に読み出して実行することにより実現される。なお、図6、図7中、S601～S604、S701～S706は各ステップを示す。

40

【0039】

CPU202は、アドレス管理モジュール1103に従ってアドレス解決を行うとき、ステップS601にて、ネットワーク上にAARPPacketパケットを送出する。本実施形態では、200msec間隔で10回、AARPPacketパケットを送出する。このAARPPacketパケットの送出は、マルチファンクション装置101が使う予定のプロトコルアドレス（仮アドレス）をネットワーク上の各AppleTalkサービスを提供しているデバイスに知らしめることを目的とする。AARPPacketパケットには、マルチファンクション装置101が使う予定のプロトコルアドレス（仮アドレス）が含まれている。

【0040】

これに対して、他のデバイスが既にそのAARPPacketパケットで通知されるプロトコルア

50

ドレスを使用していた場合、他のデバイスはAARPReplyパケットを送出し、そのプロトコルアドレスがバッティングしている旨を通知する。

#### 【0041】

次に、ステップS602において、CPU202は、ステップS601で送出したAARPProbeパケットに対して、ネットワーク上の他のデバイスから応答(AARPReply)があるか否かを判定する。そして、ステップS602で、CPU202が、他のデバイスから応答(AARPReply)があったと判定した場合には、CPU202は、仮アドレスがバッティングしていると判断して、ステップS603に処理を進める。そして、ステップS603において、CPU202は、仮アドレスを変更し、ステップS601へ処理を戻し、再度、ネットワーク上にAARPProbeパケットを送出を行う。  
10

#### 【0042】

そして、ステップS602で、CPU202が、ステップS601で送出したAARPProbeパケットに対して、ネットワーク100上の他のどのデバイスからも応答(AARPReply)がないと判定した場合には、ステップS604に処理を進める。この場合、CPU202は、ここで用いたプロトコルアドレス(どのデバイスからもAARPReplyがないプロトコルアドレス)を仮アドレスとして使用可能と判断する。

#### 【0043】

そして、ステップS604において、CPU202は、ここで用いたプロトコルアドレスを仮アドレスに決定してRAM204上に保持する。そして、ステップS701へ処理を進める。

#### 【0044】

次に、アドレス管理モジュール1103においてCPU202は、ステップS701にて、自デバイスがどのゾーンに属しているかを確認する。このとき、CPU202は、GetNetInfoのパケットをネットワーク上に送信する。なお、AppleTalkのプロトコルルータとしてのPC103は、ネットワーク番号と物理ネットワークとの対応およびゾーンの対応等の情報を管理しており、上述のGetNetInfoパケットに対するリプライ(Reply)パケットを送信する。このReplyパケットには、ネットワーク番号の範囲、デフォルトゾーンの情報等が含まれる。  
20

#### 【0045】

そして、CPU202は、ステップS702において、ステップS701で送信したGetNetInfoパケットのリプライを待機し、Replyが帰ってきたか否かを判定する。

#### 【0046】

ステップS702で、CPU202が、GetNetInfoパケットのReplyが帰ってきたと判定した場合には、該ReplyパケットをRAM204上に保持し、ステップS703に処理を遷移させる。  
30

#### 【0047】

そして、ステップS703において、CPU202は、ステップS605で決定した仮アドレスのネットワーク番号がGetNetInfoのReplyパケット中に含まれるネットワーク番号の範囲内かどうかを確認する。なお、AppleTalkのプロトコルアドレスは、ネットワーク番号、ノードID、ソケット番号等で構成されている。

#### 【0048】

そして、ステップS703で、CPU202が、上記仮アドレスのネットワーク番号がGetNetInfoのReplyパケット中に含まれるネットワーク番号の範囲外であると判定した場合には、ステップS706に処理を遷移させる。  
40

#### 【0049】

そして、ステップS706において、CPU202は、GetNetInfoのリプライパケットで示されたネットワーク番号で再度、アドレスの決定シーケンスを実行せらるよう仮アドレスを変更し、ステップS601に処理を戻す。

#### 【0050】

一方、ステップS703で、CPU202が、ステップS605で決定した仮アドレスのネットワーク番号がGetNetInfoのリプライパケット中に含まれるネットワーク番号の範囲内であると判定した場合、ステップS705に処理を遷移させる。

#### 【0051】

次に、ステップS705において、CPU202は、ステップS605で決定した仮アドレスがゾーン内で有効と判断して、ステップS605で決定した仮アドレスを正式アドレスに決定して運用を開始し、本フローチャートの処理を終了する。

#### 【0052】

一方、ステップS702で、CPU202が、ステップS701で送信したGetNetInfoパケットのReplyが帰ってこなかつたと判定した場合には、ステップS704に処理を遷移させる。

#### 【0053】

そして、ステップS704において、CPU202は、ルータが存在しないと判断し、ゾーンをデフォルトに設定し、ステップS705に処理を遷移させる。そして、ステップS705において、CPU202は、ステップS605で決定した仮アドレスを、そのまま正式アドレスに決定して運用を開始し、本フローチャートの処理を終了する。10

#### 【0054】

このような処理の流れをふまえて、AppleTalkプロトコルでは、プロトコルアドレスを決定している。

#### 【0055】

図8は、本発明のネットワークデバイスにおける第3の制御処理手順の一例を示すフローチャートであり、図5のステップS503に示した名前解決処理に対応する。なお、このフローチャートの処理は、図2に示したCPU202が、HDD207に格納されたプログラムをRAM204上に読み出して実行することにより実現される。なお、図8中、S801～S804は各ステップを示す。20

#### 【0056】

アドレス管理モジュール1103においてCPU202は、図5に図示したように、プロトコルアドレスが決まった後には名前の解決を行う。名前解決を行うとき、CPU202は、ステップS801において、自デバイスが使用する名前を保存したNBPLkUpパケットをネットワーク100上に送出し、他デバイスからの応答を待機する。

#### 【0057】

次に、ステップS802において、CPU202は、ステップS801で送出したNBPLkUpパケットのReplyパケットが帰ってきたか否かを判定する。

#### 【0058】

そして、ステップS802で、CPU202が、ステップS801で送出したNBPLkUpパケットのReplyパケットが帰ってきたと判定した場合には、該ReplyパケットをRAM204上に保持し、ステップS803に処理を遷移させる。30

#### 【0059】

この場合、CPU202は、ネットワーク上の他のデバイスがNBPLkUpパケット内の名前を使用していると判断する。そして、ステップS803において、CPU202は、名前がバッティングしているので、名前を変更し、ステップS801に処理を戻し、再確認させる。

#### 【0060】

一方、ステップS802で、CPU202が、ステップS801で送出したNBPLkUpパケットのReplyパケットが帰ってこなかつたと判定した場合には、ステップS804に処理を遷移させる。

#### 【0061】

この場合、CPU202は、ネットワーク上のどのデバイスもNBPLkUpパケット内の名前を使用していないと判断する。そして、ステップS804において、CPU202は、ステップS801で送出したNBPLkUpパケット内の名前を正式な名前に決定して、サービス提供を開始し、本フローチャートの処理を終了する。40

#### 【0062】

以上のような処理の流れで、マルチファンクション装置101は、AppleTalkを用いたプリントサービスをネットワークに提供する。

#### 【0063】

なお、Appletalkのプロトコルでは、ルータを監視する必要がある。以下、このルータを監視してプロトコルアドレスを更新する処理の流れを図9を用いて説明する。50

**【 0 0 6 4 】**

図9は、本発明のネットワークデバイスにおける第4の制御処理手順の一例を示すフローチャートであり、AppleTalkプロトコルにおいてルータ監視を行ってプロトコルアドレスの更新する処理の流れに対応する。なお、このフローチャートの処理は、図2に示したCPU202が、HDD207に格納されたプログラムをRAM204上に読み出して実行することにより実現される。なお、図9中、S901～S902は各ステップを示す。

**【 0 0 6 5 】**

アドレス管理モジュール1103においてCPU202は、ステップS901にて、ルータからのRTMPパケットを監視し、ルータからのRTMPパケットを検出する。なお、ルータからのRTMPパケットはルータ（本実施形態ではPC103）から定期的に送られてくるパケットである。このルータから定期的に送られてくるRTMPパケットを検出し、ネットワーク番号（Networkアドレス）が変更されたかどうかを判断する。10

**【 0 0 6 6 】**

ステップS902において、CPU202は、ステップS901で検出したルータからのRTMPパケットに基づいて、ネットワーク番号が変更されているか否かを判定する。ステップS902で、ネットワーク番号が変更されていないと判定した場合には、ステップS901に処理を戻し、引き続きルータからのRTMPパケットを監視する。

**【 0 0 6 7 】**

一方、ステップS902で、ネットワーク番号が変更されていると判定した場合には、図6に示したシーケンスを実行し、プロトコルアドレスの変更（更新）を行う。20

**【 0 0 6 8 】**

このようにして、デバイスは、ルータの状態を監視しながら、ネットワークにプリントサービスを提供し、印刷データ待ちとなる。

**【 0 0 6 9 】**

以下、図10を参照して、一般的なAppleTalkプロトコルを用いた印刷の流れを説明する。

**【 0 0 7 0 】**

図10は、一般的なAppleTalkプロトコルを用いた印刷の流れを示すタイミングチャートである。

**【 0 0 7 1 】**

図10に示すように、印刷を実行するPC（以下、単にPCと称する）から、NBPLkUpパケットが送出され、印刷サービスを提供しているデバイス（以下、単にデバイスと称する）の名前をネットワークから監視する。そして、デバイスは、NBPLkUpReplyとして、デバイスのプロトコルアドレスと、印刷用のソケット番号(SLS)をPCに返す。30

**【 0 0 7 2 】**

PCは、NBPLkUpReplyで返されたプロトコルアドレス、および、SLSに対して、OPENConnパケットを送出し、コネクションを生成する。そして、デバイスは、その応答OpenConnReplyをPCに返し、印刷用のコネクションが生成される。

**【 0 0 7 3 】**

そして、デバイスは、データ受信の準備ができたことを示すSendDataパケットをPCに送出する。PCはそれを受けて、印刷用のデータ（Data）をデバイスに送出する。40

**【 0 0 7 4 】**

そして、印刷するデータが終了するまで、このシーケンスを繰り返し、印刷データがなくなったところで、PCは、CloseConnパケットをデバイスに送出して、コネクションを終了する。PCはそれを受けて、その応答CloseConnReplyをPCに返す。

**【 0 0 7 5 】**

以下、図11を参照して、図1に示したマルチファンクション装置101におけるモジュール構成について説明する。

**【 0 0 7 6 】**

図11は、図1に示したマルチファンクション装置101におけるモジュール構成を示す50

概略図である。

【0077】

図11において、1103はアドレス管理モジュール（AppleTalkアドレス管理モジュール）である。1102はプロトコルスタックモジュール（AppleTalkプロトコルスタックモジュール）である。1101はプリントサービスモジュール（AppleTalkプリントサービスモジュール）である。1104は電源状態監視モジュールである。

【0078】

ネットワークから受信したパケットは、アドレス管理モジュール1103、プロトコルスタックモジュール1102で処理され、プリントサービスモジュール1101によりプリントティングモジュールに転送される。

10

【0079】

そして、このパケットは、プリントティングモジュール内で、印刷可能なビットマップデータに展開され、プリンタエンジンに送られ、印刷される。

【0080】

以上が、AppleTalkにおけるプリントサービスの処理の流れである。なお、電源状態監視モジュール1104は、省電力モードへの移行時にアドレス管理モジュール1103を停止制御する。また、電源状態監視モジュール1104は、省電力モードから通常の動作モードへの復帰時にアドレス管理モジュール1103を起動制御する。

【0081】

以下、図12を参照して、本実施形態における省電力モード（省電力状態）へ移行する際の処理及び省電力モードから通常の動作モードへ復帰した際（省電力状態が解除された際）の処理の大まかな処理の流れについて説明する。

20

【0082】

図12は、本発明のネットワークデバイスにおける第5の制御処理手順の一例を示すフローチャートであり、本実施形態における省電力モードへ移行する際の処理及び省電力モードから復帰した際の処理の大まかな処理の流れに対応する。なお、このフローチャートの処理は、図2に示したCPU202が、HDD207に格納されたプログラムをRAM204上に読み出して実行することにより実現される。なお、図12中、S1301～S1306は各ステップを示す。

【0083】

省電力モードに移行する際、電源監視モジュール1104においてCPU202は、まず、不図示のステップにて、その時点で動作していたタスクのプログラムカウンタやスタック情報、CPU202のレジスタ情報等をRAM204上に保存する。さらに、CPU202は、各デバイスユニットの終了処理を実行する。

30

【0084】

次に、ステップS1301において、CPU202は、この時点で使用しているプロトコルアドレス及び名前等の情報をRAM204上に保存し、アドレス管理モジュール1103を停止（終了）させる。次に、ステップS1302において、CPU202は、CPU202自身を停止するように制御する。即ち、CPU202を含む図3の301に示した範囲に対する電源の遮断を電源制御ユニット215に指示する。

40

【0085】

省電力モードから復帰する場合、電源制御ユニット215は、図3に示した301で囲まれた部分（CPU202を含む）に通電する。そして、通電されたCPU202は、該省電力モードから復帰した際（省電力状態が解除された際）、電源監視モジュール1104において、図示しないステップで、各デバイスユニットの初期化処理を実行する。

【0086】

次に、ステップS1303において、CPU202は、アドレス管理モジュール1103を起動する。

【0087】

そして、アドレス管理モジュール1103は、ステップS1304で、RAM204上に保存されている省電力モード移行前に使用していたプロトコルアドレスを読み出してプロトコルアドレスを解決する（省電力モードから復帰した際のアドレス解決処理）。次に、アドレス管理

50

モジュール1103は、ステップS1305において、RAM204上に保存されている省電力モード移行前に使用していた名前を読み出して名前の解決を行う（省電力モードから復帰した際の名前解決処理）。そして、ステップS1306において、CPU202は、印刷サービスを開始する。

【0088】

ここで、当該省電力モードから復帰した際の処理と図5に示した起動時の立ち上がりの処理の流れとを比較する。当該省電力モードからの復帰時の処理では、アドレス管理モジュール1103を起動した後、プロトコルスタックモジュール1102起動処理（図5のS504）とプリントサービスモジュール1101起動処理（図5のS505）を省くことができる。以下、その実現方法について、詳細に記述する。

10

【0089】

以下、図13、図14を用いて、デバイス101の省電力モードへ移行する際の処理、および、省電力モードから復帰した際の処理、電源投入された際の処理の違いを説明する。

【0090】

図13は、本発明のネットワークデバイスにおける第6の制御処理手順の一例を示すフローチャートであり、本実施形態におけるデバイスの省電力モードへ移行する際の処理に対応する。なお、このフローチャートの処理は、図2に示したCPU202が、HDD207に格納されたプログラムをRAM204上に読み出して実行することにより実現される。なお、図13中、S1401～S1405は各ステップを示す。

20

【0091】

ユーザが図4に示したユーザインターフェース部分214上のON/OFFボタン405を押下した場合や、一定時間ジョブが投入されない場合に、デバイス101は、省電力モードへ移行する。

【0092】

この際、ステップS1401で、電源状態監視モジュール1104においてCPU202は、その時点で動作していたタスク（モジュール）のプログラムカウンタやスタック情報、CPU202のレジスタ情報等をRAM204上に保存する（図12には不図示）。この情報は、省電力モードに移行する際、その時点で動作している処理を、省電力モードからの復帰時に、その状態から復帰（再開）させるためのものである。

30

【0093】

次に、ステップS1402において、CPU202は、各デバイスユニット、HDD207やスキヤナ制御201やエンジン制御206等の終了処理を実行する（図12には不図示）。このとき、I0制御部209では、タッチパネル401上の画面を暗くし、ユーザに省電力モードへ移行したことを知らせ、ON/OFFボタン405のみを有效地に制御する。

【0094】

次に、CPU202は、ステップS1403において、この時点で使用しているプロトコルアドレス及び名前等の情報をRAM204上に保存し、アドレス管理モジュール1101を停止させる（図12のステップS1301に対応）。

【0095】

次に、CPU202は、次回CPU202が起動したときに、省電力モードからの復帰であることが分かるように、電源制御ユニット215に省電力モード移行情情報をセットする。これにより、電源制御ユニット215は、この省電力モード移行情情報を内部の図示しない記憶装置に記憶する。

40

【0096】

そして、CPU202は、電源制御ユニット215に対して電源遮断を設定（指示）する。なお、上記ステップS1404、S1405は図12のステップS1302に対応する。

【0097】

この電源遮断要求を受け付けると、電源制御ユニット215は、図3で示した301で囲まれたユニット全ての電源を遮断する。これによって、CPU202の電源も遮断され、低電力が実現されることになる。なお、図3に示したように、RAM204に保持されたプログラムおよび

50

CPU関連の情報、プロトコルアドレス及び名前の情報については、301部分の電源が遮断されてもRAM204には通電されていることから、復帰情報として保持されている。従って、省電力モードからの復帰時には、これらのRAM204に保持された情報を用いて、CPU202は省電力モード移行直前に動作していたプログラムから動作することが可能になる。

#### 【0098】

次に、CPU202の電源供給時の処理について説明する。

#### 【0099】

CPU202に電源が供給される場合には、大きく2つの場合がある。1つ目は、省電力モードから復帰された場合、2つ目は、メイン電源がONにされた場合である。

#### 【0100】

省電力モードからの復帰では、ユーザが図4に示したON/OFFボタン405を押下した場合や、ネットワーク制御ユニット210にネットワークから起動要求を含むパケットが着信した場合等に、デバイス装置101は、省電力モードから復帰する。

#### 【0101】

まず、ユーザがON/OFFボタン405を押下した場合、10制御部209を介して電源制御ユニット215にPowerOn信号302が送られる。電源制御ユニット215は、そのPowerOn信号302を受けて、図3に示した301の範囲内にあるユニット全てに電源を通電し、デバイス101を省電力モードから復帰させる。これにより、CPU202が稼動状態に移行する。

#### 【0102】

同様に、ネットワークから起動要求を含むパケットが着信した場合、ネットワーク制御ユニット210から電源制御部215にPowerOn信号302が送られる。電源制御ユニット215は、そのPowerOn信号302を受けて、図3に示した301の範囲内にあるユニット全てに電源を通電し、デバイス101を省電力モードから復帰させる。これにより、CPU202が稼動状態に移行する。

#### 【0103】

そして、CPU202に電源が通電されると、CPU202は、RAM204内に保存される起動ベクタに登録されているプログラムに従って、処理を実行する。そのプログラムの処理の流れを図14に示す。

#### 【0104】

図14は、本発明のネットワークデバイスにおける第7の制御処理手順の一例を示すフローチャートであり、本実施形態におけるデバイスの電源供給時の処理に対応する。なお、このフローチャートの処理は、図2に示したCPU202が、HDD207に格納されたプログラムをRAM204上に読み出して実行することにより実現される。なお、図14中、S1501～S1505は各ステップを示す。

#### 【0105】

まず、ステップS1501において、CPU202は、電源制御ユニット215の省電力モード移行情報を読み込む（図12には不図示）。そしてステップS1502において、CPU202は、電源制御ユニット215に省電力モード移行情報が存在するか否かにより、省電力モードからの復帰（省電力モード移行していた）か否かを判定する（図12には不図示）。なお、CPU202が電源制御ユニット215に省電力モード移行情報が存在すると判断した場合には、CPU202は、省電力モードからの復帰（省電力モード移行していた）と判定する。一方、CPU202が電源制御ユニット215に省電力モード移行情報が存在しないと判断した場合には、省電力モードからの復帰でない（省電力モード移行していなかった）、即ち、メイン電源OFFからの復帰と判定する。

#### 【0106】

ステップS1502で、CPU202が、省電力モード移行していなかった（メイン電源OFFからの復帰）と判定した場合には、ステップS1506において、CPU202は、通常の起動処理を行うため、ブート処理を実行する（図12には不図示）。そして、本フローチャートの処理を終了する。このブート処理は、具体的には、HDD207を初期化し、HDD207から実行プログラムを読み出し、RAM204上に展開して、そのプログラムで各モジュールを実行し、サービス

10

20

30

40

50

を提供していく。

#### 【0107】

一方、ステップS1502で、CPU202が、省電力モード移行していたと判定した場合には、即ち、省電力モードから復帰した際、ステップS1503において、CPU202は、各ユニットの初期化処理を実行する。このとき、I/O制御部209はタッチパネル画面401を表示し省電力モードから通常モードへ復帰したことをユーザに知らしめる（図12には不図示）。

#### 【0108】

次に、ステップS1504において、CPU202は、AppleTalkのアドレス管理モジュール1102を起動してアドレス管理開始を通知する（図12のステップS1303に対応）。これにより、後述する図15、図16に示す処理が実行される。

10

#### 【0109】

その後、ステップS1505において、CPU202は、RAM204に保持しておいたプログラムカウンタ、スタック情報、CPUレジスタ等をCPU202に戻し、省電力モード移行前の状態から処理を再開するように制御する（復帰制御する）。そして、本フローチャートの処理を終了する。これにより、印刷サービスが再開される（図12のステップS1306に対応）。

#### 【0110】

なお、先に説明したように、RAM204は省電力モードとなっている間も電源が通電されていることから、プロトコルアドレス、名前、およびプログラム等、メモリ上に展開されたデータは省電力モードとなっている間も保持されつづけることになる。

#### 【0111】

以下、図15、図16を用いて、省電力モードからの復帰時のアドレス解決処理、省電力モードからの復帰時の名前解決処理の流れについて説明する。

20

#### 【0112】

図15は、本発明のネットワークデバイスにおける第8の制御処理手順の一例を示すフローチャートであり、本実施形態におけるデバイスの省電力モードからの復帰した際のアドレス解決処理（図12のステップS1304）に対応する。なお、このフローチャートの処理は、図2に示したCPU202が、HDD207に格納されたプログラムをRAM204上に読み出して実行することにより実現される。なお、図15中、S1601～S1606は各ステップを示す。

#### 【0113】

図14のステップS1504で、アドレス管理モジュール1103にアドレス管理開始を通知すると、アドレス管理モジュール1103においてCPU202は、本フローチャートの処理を開始する。この処理で、CPU202は、省電力モード移行していた間に保持していたプロトコルアドレスがそのまま使用できるか、名前が使用できるかどうかを判断し、使用できる場合にはそのまま使用しつづける。

30

#### 【0114】

省電力モードから復帰した際、まず、ステップS1601において、CPU202は、省電力モード移行前に使用していたプロトコルアドレスをRAM204から読み出す。そして、CPU202は、該省電力モード移行前に使用していたプロトコルアドレスが使用可能かどうかを、ネットワークにARPProbeパケットを送出することで確認する。

#### 【0115】

CPU202は、CPU202が省電力モード移行していた間に、ほかのPCがデバイスの使用していたプロトコルアドレスを使用てしまっているかどうかを、ARPProbeパケットのレスポンスの有無で確認する。

40

#### 【0116】

ステップS1602において、CPU202は、ARPProbeパケットにレスポンス（AARPReply）が有るか否かを判定し、ARPProbeパケットにレスポンスがないと判定した場合には、ステップS1603に処理を進める。この場合、CPU202は、どのデバイスも、そのプロトコルアドレスを使用していなかったと判断する。そして、ステップS1603において、CPU202は、そのままそのプロトコルアドレスを継続して使用するかどうかを判断するため、GNI（GetNetInfo）パケットを送出してルータの情報を確認する。

50

**【 0 1 1 7 】**

CPU202は、省電力モードとなっていた間にルータの情報が変わっていたかどうかを判断する。CPU202は、GetNetInfoパケットのリプライを待機し、GetNetInfoパケットのReplyに基づいてネットワーク番号を確認する。そして、ステップS1604において、CPU202は、省電力モード移行前に使用していたアドレスのネットワーク番号がGetNetInfoのReplyパケット中に含まれるネットワーク番号の範囲内かどうかを判定する。

**【 0 1 1 8 】**

ステップS1604で、CPU202が、範囲内であると判定した場合には、省電力モード移行前に使用していたプロトコルアドレスをそのまま継続して使用するように決定し、本フローチャートの処理を終了する。そして、図16に示す省電力モードからの復帰時の名前解決処理へ移行させる。10

**【 0 1 1 9 】**

ステップS1604で、CPU202が、範囲内でないと判定した場合（GetNetInfoパケットのReplyが帰って来なかつた場合も含む）には、ステップS1606に処理を遷移させる。

**【 0 1 2 0 】**

また、ステップS1602で、CPU202が、ARPProbeパケットにレスポンス（AARPReply）が有ったと判定した場合にも、ステップS1606に処理を進める。

**【 0 1 2 1 】**

ステップS1606では、CPU202は、再度アドレスを取り直すように制御し（図6、図7に示した処理と同様）、アドレスを取り直した後、本フローチャートの処理を終了して、図16に示す省電力モードからの復帰時の名前解決処理へ移行させる。20

**【 0 1 2 2 】**

図16は、本発明のネットワークデバイスにおける第9の制御処理手順の一例を示すフローチャートであり、本実施形態におけるデバイスの省電力モードからの復帰した際の名前解決処理（図12のステップS1305）に対応する。なお、このフローチャートの処理は、図2に示したCPU202が、HDD207に格納されたプログラムをRAM204上に読み出して実行することにより実現される。なお、図16中、S1701～S1704は各ステップを示す。

**【 0 1 2 3 】**

図15に示した処理でアドレスが確定した後、本フローチャートの処理を開始する。まず、ステップS1701において、CPU202は、省電力モード移行前に使用していた名前をRAM204から読み出す。そして、CPU202は、該省電力モード移行前に使用していた名前が継続して使用できるかどうかをネットワークにNBPLkUpパケットを送出して確認する。30

**【 0 1 2 4 】**

ステップS1702において、CPU202は、NBPLkUpパケットにレスポンス（NBPLkUpReply）が有るか否かを判定する。そして、ステップS1702で、CPU202が、NBPLkUpパケットにレスポンス（NBPLkUpReply）がないと判定した場合には、ステップS1703に処理を進める。この場合、CPU202は、省電力モードに移行していた間に同じ名前を使用するものがいなかったと判断して、ステップS1703において、CPU202は、省電力モード移行前に使用していた名前をそのまま使用を継続するように決定する。そして、本フローチャートの処理を終了する。40

**【 0 1 2 5 】**

一方、ステップS1702で、CPU202が、NBPLkUpパケットにレスポンス（NBPLkUpReply）が有ったと判定した場合には、ステップS1704に処理を進める。この場合、CPU202は、省電力モード中に同じ名前を使用するものがいたと判断して、ステップS1704において、CPU202は、再度、名前を取り直すように制御し（図8に示した処理と同様）、名前を取り直す。そして、本フローチャートの処理を終了する。

**【 0 1 2 6 】**

以上のようにして、CPUが電源遮断されていた間（省電力モード中）にプロトコルアドレスや名前が他のデバイスに使用されてしまった場合について対応する。なお、プロトコルスタックモジュールや、プリントサービスモジュールは、省電力モードからの復帰後も50

省電力モード移行前と変わらず、RAM204上に展開されており、プログラムとして変わらず動作可能である。よって、アドレス管理モジュール1103のみを起動させることによって、省電力モードからの復帰後もAppleTalkの印刷サービスを提供することができる。

#### 【0127】

以下、本実施形態における、全体の処理の流れについて説明する。

#### 【0128】

まず、電源ONから印刷までの処理の流れについて説明する。

#### 【0129】

デバイスのメイン電源が投入されると、図14で示したような流れで、CPU202は立ち上がりの処理を行う。デバイスのメイン電源の投入の場合には、ステップS1502で判断し、ブート処理が実行される。ブート処理によって、プログラムが展開後、各サービスが実行可能となっていく。10

#### 【0130】

AppleTalkの印刷サービスにおいては、図5に示したように、アドレス管理モジュール1103が起動し、アドレス管理モジュールの中で、プロトコルアドレス解決処理、名前の解決処理、が行われる。これらの処理は、図6、図7で説明したような処理の流れで実現される。11

#### 【0131】

その後、アドレス管理モジュール1103は、名前の解決処理を図8で説明したような処理の流れで行う。プロトコルスタックモジュール1102が起動し、プリントサービスモジュール1101が起動すると、デバイスはAppleTalkの印刷サービスを開始することが可能となる。なお、印刷サービスを提供している間、アドレス管理モジュール1103は、図9に図示したような処理の流れで、ルータから送出されるパケットを監視している。20

#### 【0132】

また、PCからの印刷要求が発生すると、図10に図示したような処理の流れで、プリントサービスモジュール1101はPCからデータを受信し、プリントティングモジュールに印刷データを転送して、印刷が実行される。一般的にはこのような処理の流れで、AppleTalkからの印刷が実行される。21

#### 【0133】

次に省電力モードに移行し、復帰した際の全体の処理の流れについて説明する。30

#### 【0134】

一定期間、コピーやプリントジョブが発生しない、または、ユーザがパネルのON/OFFボタン405から省電力モードをONにした場合には、デバイスは省電力モードに移行する。このときの処理の流れは、図12のステップS1301,1302に示したとおりで、電源状態監視モジュール1104は、省電力モードへ移行する際の処理を行い、アドレス管理モジュール1103への停止処理を行い、CPU202を停止させる。この詳細処理は図13に図示したとおりである。アドレス管理モジュール1103は、電源状態監視モジュール1104から停止要求を受け付けると、ルータの監視（プロトコルアドレスの更新処理）をとめる。このため、CPU202が電源遮断されている間には（省電力モード状態では）、デバイス101が使用していたプロトコルアドレスや、名前を他のデバイスが使用することが可能となる。40

#### 【0135】

このような省電力モード状態で、PCからAppleTalkの印刷要求が来たときの処理の流れについて、以下、図17を用いて説明する。

#### 【0136】

図17は、デバイスが省電力モード状態でPCからAppleTalkの印刷要求が来たときの処理の流れを示すタイミングチャートである。

#### 【0137】

図17に示すように、省電力モード状態においても、ネットワーク制御ユニット210には通電されていることから、ネットワーク制御ユニット210ではネットワークパケットを受信することが可能である。なお、ネットワーク制御ユニット210には予め、決まったフ50

オーマットのパケットを受信した場合に電源制御ユニット215に対して、PowerOn信号302を送出するように設定してある。本実施形態では、NBPLkUpパケットを受信した場合に、電源制御ユニット215に対して、PowerOn信号302を送出するよう設定している。なお、これ以外のAppleTalkのパケットで省電力モードから復帰するように設定してしまうと、ネットワーク上に流れるデバイスと無関係なAppletalkパケットにより、デバイスがたびたび起動してしまう。このため、このNBPLkUpパケットのみのパターンで、電源制御ユニット215に対してPowerOn信号302を送出するようにネットワーク制御ユニット210に設定する。

#### 【0138】

これにより、NBPLkUpパケットをネットワーク制御ユニット210が受信すると、ネットワーク制御ユニット210は、PowerOn信号302を電源制御ユニット215に送出する。このPowerOn信号302を受信した電源制御ユニット215が、図3に示した301で囲まれた各ユニットの電源をONにすることで、省電力モードからの復帰処理1201が行われる。10

#### 【0139】

この省電力モードからの復帰処理1201は、図14に図示したような処理の流れで行われる。図14のステップS1502で、省電力モードからの復帰したと判断した場合には、各デバイスの初期化処理を実行し、アドレス監視モジュール1103に開始を通知する。そして、CPU202はRAM204からプログラムカウンタを復帰させて、省電力モード移行時に停止させたプログラムカウンタから処理を再開することが可能となる。20

#### 【0140】

アドレス管理モジュール1103が処理を開始する場合には、アドレス解決処理1202が行われる。このアドレス解決処理1202は、図15、図16で示したような処理の流れで行われ、今まで使用していたプロトコルアドレスや名前がそのまま継続して使用可能かどうかを判断し、継続処理を行う。20

#### 【0141】

こうして、プロトコルアドレスと名前を再度確認した後、デバイスは、PCに対して、NBPLkUpReplyを送出する。省電力モードとなっている間にプロトコルアドレスが変わってしまっても、新しいプロトコルアドレスをNBPLkUpReplyに返すことによって、それ以後、PCはデバイスにコネクションを生成し、印刷することが可能となる。30

#### 【0142】

このように、省電力モードに移行する際には、アドレス管理モジュール1103を停止させ、省電力モードから復帰した場合には、アドレス管理モジュール1103を動作させて、プロトコルアドレスを解決する。これによって、CPU電源が切れてしまうような省電力モードの環境下でも、AppleTalkのサービスを提供することが可能となる。30

#### 【0143】

以上、説明したように、本実施形態によれば、省電力モードからの復帰時に、AppleTalkのサービスが有効な場合でも、CPUを停止させ、省電力モードに移行することが可能になる。

#### 【0144】

##### 〔第2実施形態〕

上記第1実施形態以外にも以下のようなシーケンスを実現することによっても対応することができる。以下、この実施形態について図18を用いて説明する。40

#### 【0145】

図18は、本発明のネットワークデバイスにおける第10の制御処理手順の一例を示すフローチャートであり、本発明の第2実施形態におけるデバイスの省電力モード移行時の処理及び省電力モードからの復帰時の処理に対応する。なお、このフローチャートの処理は、図2に示したCPU202が、HDD207に格納されたプログラムをRAM204上に読み出して実行することにより実現される。なお、図18中、S1801～S1808は各ステップを示す。

#### 【0146】

まず、省電力モードに移行する場合、CPU202は、ステップS1801において、動作してい50

る全プログラムを終了(Shutdown)させる。

【0147】

そして、省電力モードから復帰したときに、CPU202は、ステップS1802において、全プログラムをロードしなおし、ステップS1803において、順次プログラムを活性化させる。次に、CPU202は、ステップS1804において、プロトコルアドレスを解決する処理を行い、ステップS1805において、名前を解決する処理を行い。さらに、CPU202は、ステップS1806において、プロトコルスタックモジュール1102を起動し、ステップS1807において、プリントサービスモジュール1103を起動する。そして、ステップS1808において、CPU202は、印刷サービスを開始する。

【0148】

なお、このようなシーケンスをとった場合、省電力モードに移行するときに、動作している全プログラムを終了させるため、第1実施形態で示したシーケンスに比べると、省電力モード移行に時間がかかってしまう。また、省電力モードからの復帰時に、全プログラムをロードしなおすために、この場合もはやり、第1実施形態で示したシーケンスに比べると、省電力モードからの復帰に時間がかかってしまう。

【0149】

〔第3実施形態〕

また、別の実施形態として、以下のようなシーケンスを実現することによっても対応することができる。以下、この実施形態について図19を用いて説明する。

【0150】

図19は、本発明のネットワークデバイスにおける第11の制御処理手順の一例を示すフローチャートであり、本発明の第3実施形態におけるデバイスの省電力モード移行時の処理及び省電力モードからの復帰時の処理に対応する。なお、このフローチャートの処理は、図2に示したCPU202が、HDD207に格納されたプログラムをRAM204上に読み出して実行することにより実現される。なお、図19中、S1901～S1908は各ステップを示す。

【0151】

まず、省電力モードに移行する場合、CPU202は、ステップS1901において、動作しているAppleTalk関連モジュール1101～1103のみを全て終了(Shutdown)させる。

【0152】

そして、省電力モードから復帰したときに、CPU202は、ステップS1902において、AppleTalk関連モジュールのみをロードしなおし、ステップS1903において、順次プログラムを活性化させる。

【0153】

次に、CPU202は、ステップS1904において、プロトコルアドレスを解決する処理を行い、ステップS1905において、名前を解決する処理を行い。さらに、CPU202は、ステップS1806において、プロトコルスタックモジュール1102を起動し、ステップS1807において、プリントサービスモジュール1103を起動する。そして、ステップS1808において、CPU202は、印刷サービスを開始する。

【0154】

なお、このようなシーケンスをとった場合、省電力モードに移行するときに、AppleTalk関連モジュール1101～1103のみを終了させるため、第2実施形態(図18)で示したシーケンスに比べると、省電力モード移行に時間は短くなる。

【0155】

また、省電力モードからの復帰時に、AppleTalk関連プログラム1101～1103のみをロードしなおすために、はやり、第2実施形態(図18)で示したシーケンスに比べると、省電力モードからの復帰時間は短くなる。

【0156】

なお、第2,3実施形態(図18,図19)に示したシーケンスに比べて、第1実施形態に説明したシーケンスによれば、格段に省電力モードの移行、復帰時間が短くなる。また同時に、AppleTalkのような動的なプロトコルアドレスを使用する環境下でも、CPUの電

10

20

30

40

50

源を遮断するような省電力モードに移行することが可能となる。

【0157】

〔第4実施形態〕

上記第1～3実施形態に示したAppleTalkのプロトコルアドレスと同様にTCP/IPのIPアドレスにおいても、同様のシーケンスでアドレス管理することができる。

【0158】

TCP/IPのIPアドレスの場合には、近年DHCPを用いた動的なIPアドレス取得が用いられている。DHCPでは、サーバからデバイスがIPアドレスを取得したのち、定期的にサーバにそのIPアドレスを使用し続けてよいかどうかをサーバに問い合わせている。

【0159】

本実施形態のアドレス管理モジュールが、DHCPプロトコルをサポートするように構成する。即ち、省電力モードへ移行する際に、この時点で使用していたIPアドレスをRAM204に保存させておく。そして、省電力モード中には、サーバへの問い合わせを行わない。そして、省電力モードからの復帰時に、省電力モード移行前に使用していたIPアドレス（RAM202に保持されている）がそのまま継続して使用可能かどうかをDHCPサーバに問い合わせるように構成する。このような構成によって、DHCP等のIPアドレスが動的に変更する可能性があるようなプロトコルの場合でも、AppleTalkプロトコルの場合と同様に、CPUを停止させた省電力モードに移行することが可能となる。そして、省電力モードからの復帰時も問題なく動作することができる。

【0160】

また、AppleTalk、TCP/IP以外のプロトコルであって、プロトコルアドレスが動的に変更する可能性があるようなプロトコルの場合、同様に、CPUを停止させた省電力モードに移行することが可能となる。そして、省電力モードからの復帰時も問題なく動作することができる。

【0161】

上記各実施形態では、マルチファンクション装置を例にして説明した。しかし、所定の通信プロトコルに対応する動的なプロトコルアドレスを用いてネットワークを介した通信が可能なネットワークデバイスであれば、どのようなデバイスにも本発明は適用可能である。

【0162】

以上、本発明の実施形態について示したが、本発明は、例えば、システム、装置、方法、プログラムもしくは記憶媒体等としての実施態様をとることが可能である。具体的には、複数の機器から構成されるシステムに適用しても良いし、また、一つの機器からなる装置に適用しても良い。

【0163】

以上示したように、CPU202は、省電力状態（省電力モード）に移行する際、その時点で使用しているアドレスをRAM204に記憶させる。また、CPU202は、省電力状態が解除された際、RAM204に記憶されている前記アドレスが使用可能かどうか判断する。そして、該アドレスが使用可能な場合、該アドレスを使用して処理を再開するように制御する。これにより、AppleTalkやTCP/IPのDHCPのようにアドレスが動的に変化する可能性のある環境下で動作するネットワークデバイスであっても、CPUへの電源を遮断した省電力モードに移行して省電力を実現できる。

【0164】

以下、図20に示すメモリマップを参照して、本発明に係るネットワーク装置で読み取り可能な各種データ処理プログラムを格納する記憶媒体のメモリマップの構成について説明する。

【0165】

図20は、本発明に係るネットワーク装置で読み取り可能な各種データ処理プログラムを格納する記憶媒体（記録媒体）のメモリマップを説明する図である。

【0166】

10

20

30

40

50

なお、特に図示しないが、記憶媒体に記憶されるプログラム群を管理する情報、例えばバージョン情報、作成者等も記憶され、かつ、プログラム読み出し側のOS等に依存する情報、例えばプログラムを識別表示するアイコン等も記憶される場合もある。

#### 【0167】

さらに、各種プログラムに従属するデータも上記ディレクトリに管理されている。また、各種プログラムをコンピュータにインストールするためのプログラムや、インストールするプログラムが圧縮されている場合に、解凍するプログラム等も記憶される場合もある。

#### 【0168】

本実施形態における図5、図6、図7、図8、図9、図12、図13、図14、図15、図16、図18、図19に示す機能が外部からインストールされるプログラムによって、ホストコンピュータにより遂行されていてもよい。そして、その場合、CD-ROMやフラッシュメモリやFD等の記憶媒体により、あるいはネットワークを介して外部の記憶媒体から、プログラムを含む情報群を出力装置に供給される場合でも本発明は適用されるものである。

#### 【0169】

以上のように、前述した実施形態の機能を実現するソフトウェアのプログラムコードを記録した記憶媒体を、システムあるいは装置に供給する。そして、そのシステムあるいは装置のコンピュータ（またはCPUやMPU）が記憶媒体に格納されたプログラムコードを読み出し実行することによっても、本発明の目的が達成されることとは言うまでもない。

#### 【0170】

この場合、記憶媒体から読み出されたプログラムコード 자체が本発明の新規な機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

#### 【0171】

従って、プログラムの機能を有していれば、オブジェクトコード、インタプリタにより実行されるプログラム、OSに供給するスクリプトデータ等、プログラムの形態を問わない。

#### 【0172】

プログラムを供給するための記憶媒体としては、例えばフレキシブルディスク、ハードディスク、光ディスク、光磁気ディスク、MO、CD-ROM、CD-R、CD-RW、磁気テープ、不揮発性のメモリカード、ROM、DVDなどを用いることができる。

#### 【0173】

この場合、記憶媒体から読み出されたプログラムコード 자체が前述した実施形態の機能を実現することになり、そのプログラムコードを記憶した記憶媒体は本発明を構成することになる。

#### 【0174】

その他、プログラムの供給方法としては、クライアントコンピュータのブラウザを用いてインターネットのホームページに接続し、該ホームページから本発明のプログラムそのものをハードディスク等の記憶媒体にダウンロードすることによっても供給できる。また、該ホームページから圧縮され自動インストール機能を含むファイルをハードディスク等の記憶媒体にダウンロードすることによっても供給できる。また、本発明のプログラムを構成するプログラムコードを複数のファイルに分割し、それぞれのファイルを異なるホームページからダウンロードすることによっても実現可能である。つまり、本発明の機能処理をコンピュータで実現するためのプログラムファイルを複数のユーザに対してダウンロードさせるWWWサーバやFTPサーバ等も本発明の請求項に含まれるものである。

#### 【0175】

また、本発明のプログラムを暗号化してCD-ROM等の記憶媒体に格納してユーザに配布する。さらに、所定の条件をクリアしたユーザに対し、インターネットを介してホームページから暗号化を解く鍵情報をダウンロードさせる。さらに、その鍵情報を使用することに

10

20

30

40

50

より暗号化されたプログラムを実行してコンピュータにインストールさせて実現することも可能である。

【0176】

また、コンピュータが読み出したプログラムコードを実行することにより、前述した実施形態の機能が実現されるだけでなく、以下のような構成も含まれることは言うまでもない。例えば、そのプログラムコードの指示に基づき、コンピュータ上で稼働しているOS(オペレーティングシステム)等が実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0177】

さらに、記憶媒体から読み出されたプログラムコードを、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込む。そして、該メモリに書き込まれたプログラムコードの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPU等が実際の処理の一部または全部を行い、その処理によって前述した実施形態の機能が実現される場合も含まれることは言うまでもない。

【0178】

また、本発明は、複数の機器から構成されるシステムに適用しても、1つの機器からなる装置に適用してもよい。また、本発明は、システムあるいは装置にプログラムを供給することによって達成される場合にも適応できることは言うまでもない。この場合、本発明を達成するためのソフトウェアによって表されるプログラムを格納した記憶媒体を該システムあるいは装置に読み出すことによって、そのシステムあるいは装置が、本発明の効果を享受することが可能となる。

【0179】

本発明は上記実施形態に限定されるものではなく、本発明の趣旨に基づき種々の変形(各実施形態の有機的な組合せを含む)が可能であり、それらを本発明の範囲から排除するものではない。

【0180】

本発明の様々な例と実施形態を示して説明したが、当業者であれば、本発明の趣旨と範囲は、本明細書内の特定の説明に限定されるのではない。

【0181】

以上説明したように、省電力モードからの復帰時に、AppleTalkのアドレス解決モジュールのみを処理させることによって、AppleTalkのサービスが有効な場合でも、CPUを停止させ、省電力モードに移行することが可能になる。

【0182】

また、DHCPなど、IPアドレスが動的に変更する可能性があるようなプロトコルの場合でも、同様に、CPUを停止させ、省電力モードに移行することが可能となる。

【0183】

従って、CPUを停止させるような省電力モード環境下でも、AppleTalkやDHCP等のような動的プロトコルアドレスを使用するプロトコルで、省電力モードへ移行できるようなシステムを提供することができる。

【0184】

なお、上述した各実施形態およびその変形例を組み合わせた構成も全て本発明に含まれるものである。

【図面の簡単な説明】

【0185】

【図1】本発明の一実施形態を示すネットワークデバイス装置として、ネットワーク接続機能を持つマルチファンクション装置がLANに接続されているシステムの概略図である。

【図2】図1に示したマルチファンクション装置の概略を表すブロック図である。

【図3】図2に示した電源制御ユニットによって、どのユニットの電源を遮断するかを示した図である。

【図4】図2に示したユーザインターフェース部分の構成を説明する平面図である。

10

20

30

40

50

【図5】本発明のネットワークデバイスにおける第1の制御処理手順の一例を示すフローチャートである。

【図6】本発明のネットワークデバイスにおける第2の制御処理手順の一例を示すフローチャートである。

【図7】本発明のネットワークデバイスにおける第3の制御処理手順の一例を示すフローチャートである。

【図8】本発明のネットワークデバイスにおける第4の制御処理手順の一例を示すフローチャートである。

【図9】本発明のネットワークデバイスにおける第5の制御処理手順の一例を示すフローチャートである。

【図10】一般的なAppleTalkプロトコルを用いた印刷の流れを示すタイミングチャートである。

【図11】図1に示したマルチファンクション装置101におけるモジュール構成を示す概略図である。

【図12】本発明のネットワークデバイスにおける第6の制御処理手順の一例を示すフローチャートである。

【図13】本発明のネットワークデバイスにおける第7の制御処理手順の一例を示すフローチャートである。

【図14】本発明のネットワークデバイスにおける第8の制御処理手順の一例を示すフローチャートである。

【図15】本発明のネットワークデバイスにおける第9の制御処理手順の一例を示すフローチャートである。

【図16】本発明のネットワークデバイスにおける第10の制御処理手順の一例を示すフローチャートである。

【図17】デバイスが省電力モードでPCからAppleTalkの印刷要求が来たときの処理の流れを示すタイミングチャートである。

【図18】本発明のネットワークデバイスにおける第11の制御処理手順の一例を示すフローチャートである。

【図19】本発明のネットワークデバイスにおける第12の制御処理手順の一例を示すフローチャートである。

【図20】本発明に係るネットワーク装置で読み取り可能な各種データ処理プログラムを格納する記憶媒体(記録媒体)のメモリマップを説明する図である。

#### 【符号の説明】

##### 【0186】

1 0 0	ネットワーク
1 0 1	マルチファンクション装置
1 0 2	PC
1 0 3	PC(ルータ)
2 0 2	CPU
2 0 3	ROM
2 0 4	RAM
2 0 7	HDD
2 1 0	ネットワーク制御ユニット210
2 1 5	電源制御ユニット
1 1 0 1	プリントサービスモジュール
1 1 0 2	プロトコルスタッキモジュール
1 1 0 3	アドレス管理モジュール
1 1 0 4	電源状態監視モジュール

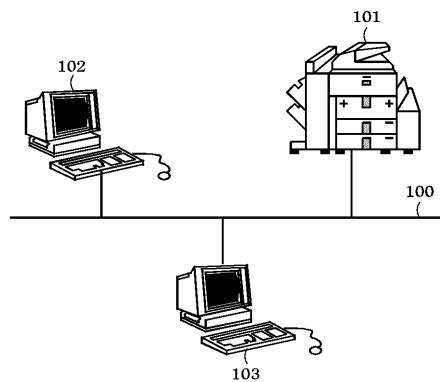
10

20

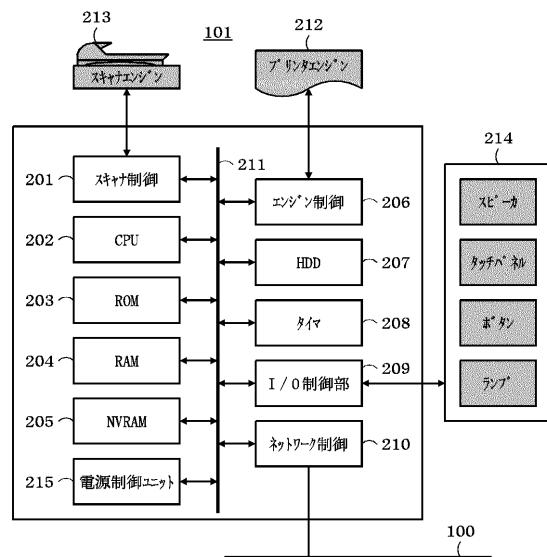
30

40

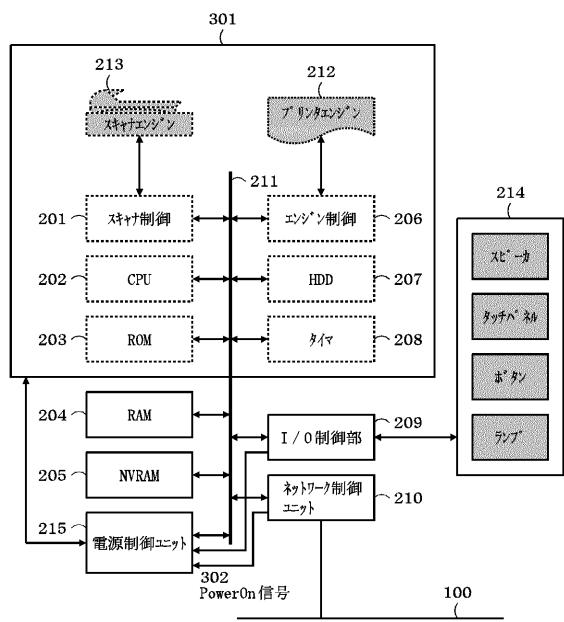
【図1】



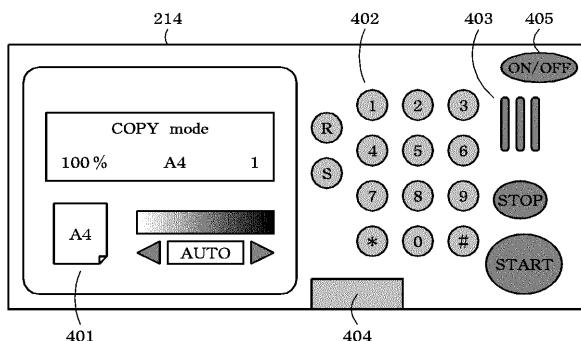
【図2】



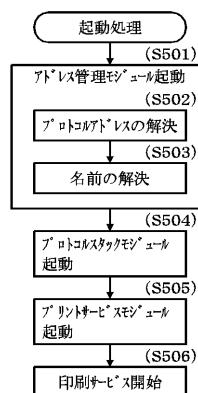
【図3】



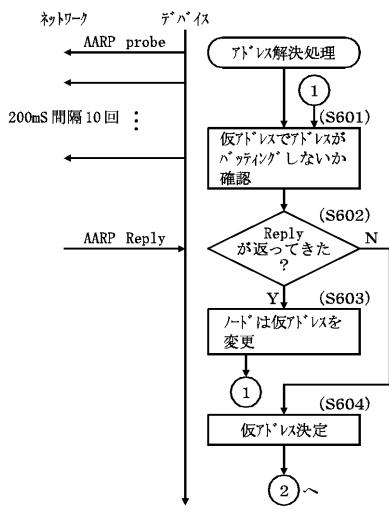
【図4】



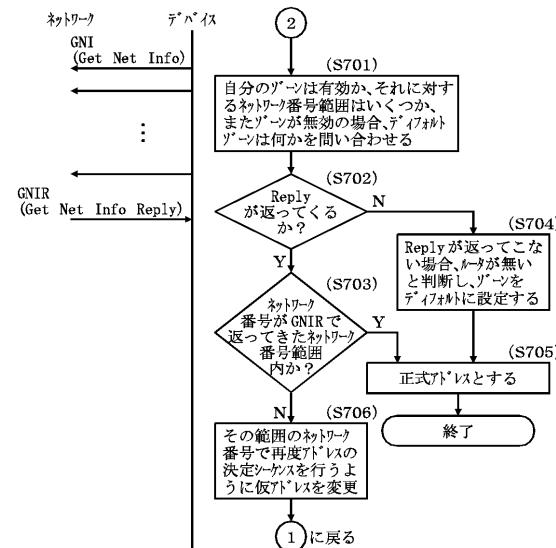
【図5】



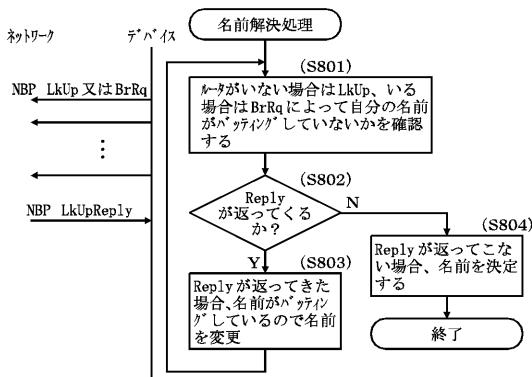
【図6】



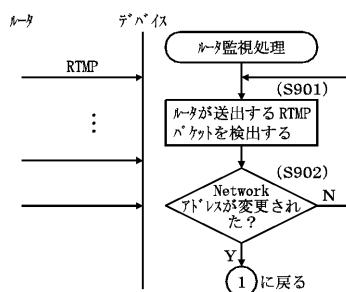
【 図 7 】



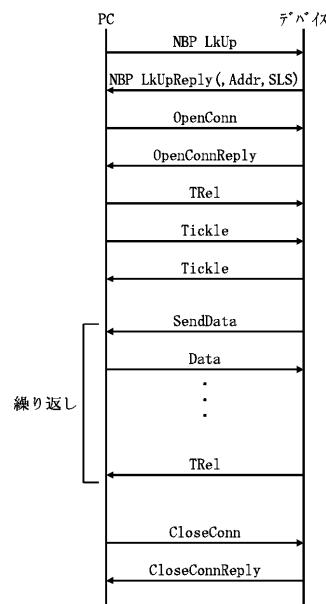
【 図 8 】



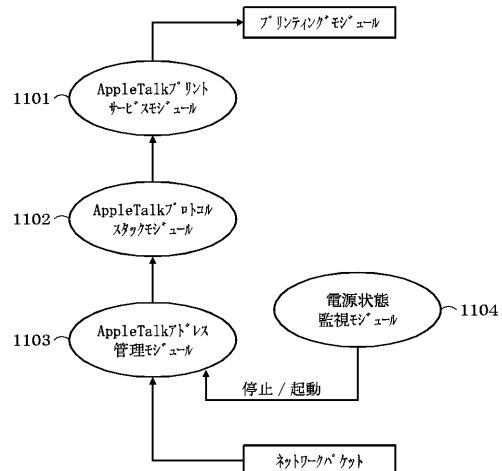
【 四 9 】



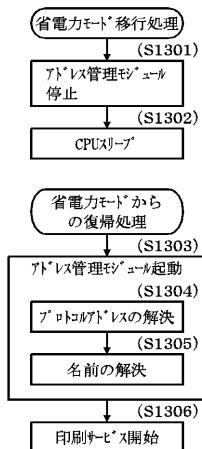
【 図 1 0 】



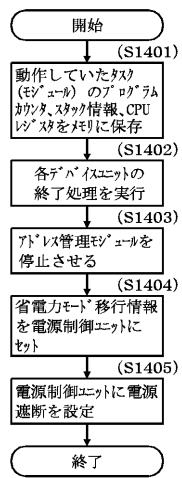
【図11】



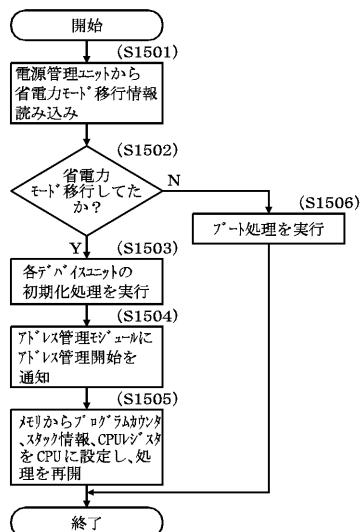
【図12】



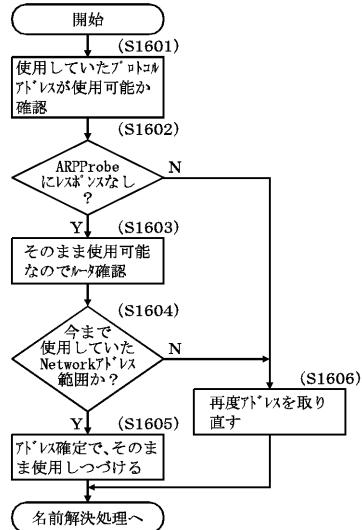
【図13】



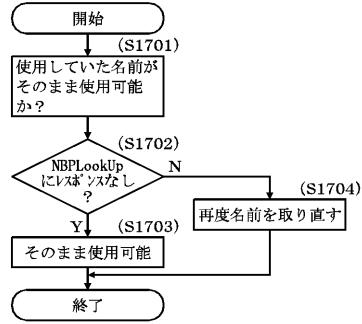
【図14】



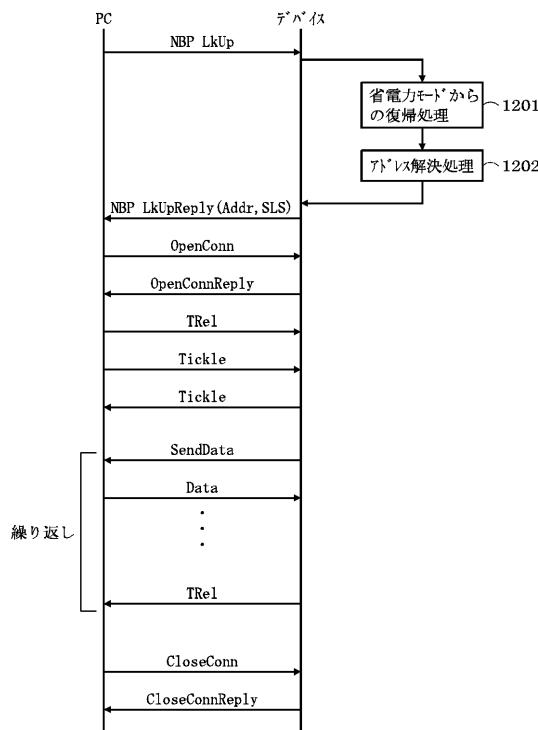
【図15】



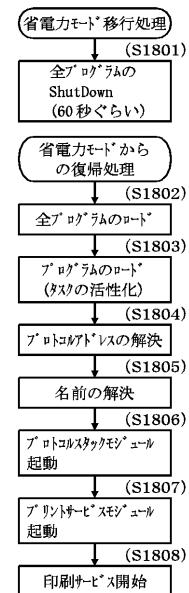
【図16】



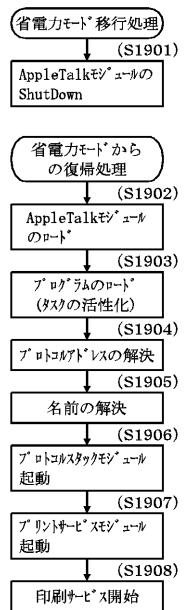
【図17】



【図18】



【図19】



【図20】

FD/CD-ROM等の記憶媒体	
ディレクトリ情報	
第1のデータ処理プログラム	図5に示すフローチャートのステップに対応するプログラムコード群
第2のデータ処理プログラム	図6、図7に示すフローチャートのステップに対応するプログラムコード群
第3のデータ処理プログラム	図8に示すフローチャートのステップに対応するプログラムコード群
第4のデータ処理プログラム	図9に示すフローチャートのステップに対応するプログラムコード群
第5のデータ処理プログラム	図10に示すフローチャートのステップに対応するプログラムコード群
第6のデータ処理プログラム	図11に示すフローチャートのステップに対応するプログラムコード群
第7のデータ処理プログラム	図12に示すフローチャートのステップに対応するプログラムコード群
第8のデータ処理プログラム	図13に示すフローチャートのステップに対応するプログラムコード群
第9のデータ処理プログラム	図14に示すフローチャートのステップに対応するプログラムコード群
第10のデータ処理プログラム	図15に示すフローチャートのステップに対応するプログラムコード群
第11のデータ処理プログラム	図16に示すフローチャートのステップに対応するプログラムコード群
	記憶媒体のメモリマップ

---

フロントページの続き

(51)Int.Cl.

F I

テーマコード(参考)

H 04N 1/00 C  
H 04N 1/00 107Z