

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第3904054号
(P3904054)

(45) 発行日 平成19年4月11日(2007.4.11)

(24) 登録日 平成19年1月19日(2007.1.19)

(51) Int.C1.

F 1

G 06 F 3/12 (2006.01)
B 41 J 29/38 (2006.01)G 06 F 3/12
G 06 F 3/12
B 41 J 29/38C
M
Z

請求項の数 12 (全 13 頁)

(21) 出願番号 特願2000-32739 (P2000-32739)
 (22) 出願日 平成12年2月4日 (2000.2.4)
 (65) 公開番号 特開2001-216114 (P2001-216114A)
 (43) 公開日 平成13年8月10日 (2001.8.10)
 審査請求日 平成16年4月13日 (2004.4.13)

(73) 特許権者 000002369
 セイコーエプソン株式会社
 東京都新宿区西新宿2丁目4番1号
 (74) 代理人 100096703
 弁理士 横井 俊之
 (72) 発明者 大石 敏昭
 長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内
 (72) 発明者 尾形 裕保
 長野県諏訪市大和3丁目3番5号 セイコーエプソン株式会社内
 審査官 田中 友章

最終頁に続く

(54) 【発明の名称】用紙排出プログラムを記録した媒体、データ識別プログラムを記録した媒体、用紙排出装置、データ識別装置、用紙排出方法およびデータ識別方法

(57) 【特許請求の範囲】

【請求項 1】

印刷ジョブがスプーラによってキャンセルされたときに印刷装置にて印刷用紙を排出させる用紙排出プログラムを記録した媒体であって、

上記スプーラが出力するスプールデータ中の後尾から所定サイズのデータを格納するスプールデータ参照機能と、

上記スプーラによるキャンセル時に起動され、上記スプールデータ参照機能により格納されているデータを参照して所定態様のデータの有無を検出するデータ検出機能と、

同データ検出機能によって所定態様のデータが検出されなかったときに上記印刷装置に対して用紙排出コマンドを出力する用紙排出コマンド出力機能とをコンピュータに実現させ、

上記所定サイズは、上記スプーラが出力する最大のデータサイズよりも小さいことを特徴とする用紙排出プログラムを記録した媒体。

【請求項 2】

上記請求項1に記載の用紙排出プログラムを記録した媒体であって、

上記スプールデータ参照機能は、上記所定態様データの検出に必要なビット数のデータを参照することを特徴とする用紙排出プログラムを記録した媒体。

【請求項 3】

上記請求項1または請求項2のいずれかに記載の用紙排出プログラムを記録した媒体であって、

10

20

上記スプールデータ参照機能は上記格納されているデータを用いて、分割された上記所定態様のデータを復元可能であることを特徴とする用紙排出プログラムを記録した媒体。

【請求項 4】

上記請求項 1 ~ 請求項 3 のいずれかに記載の用紙排出プログラムを記録した媒体において、

上記データ検出機能は、印刷ジョブの終了を示すデータを検出することを特徴とする用紙排出プログラムを記録した媒体。

【請求項 5】

印刷ジョブがスプーラによってキャンセルされたときに印刷装置にて印刷用紙を排出させる用紙排出装置であって、

上記スプーラが¹⁰出力するスプールデータ中の後尾から所定サイズのデータを格納するスプールデータ参照手段と、

上記スプーラによるキャンセル時に起動され、上記スプールデータ参照手段により格納されているデータを参照して所定態様のデータの有無を検出するデータ検出手段と、

同データ検出手段によって所定態様のデータが検出されなかったときに上記印刷装置に対して用紙排出コマンドを出力する用紙排出コマンド出力手段とを有し、上記所定サイズは、上記スプーラが出力する最大のデータサイズよりも小さいことを特徴とする用紙排出装置。

【請求項 6】

上記請求項 5 に記載の用紙排出装置であって、

20

上記スプールデータ参照手段は、上記所定態様データの検出に必要なビット数のデータを参照することを特徴とする用紙排出装置。

【請求項 7】

上記請求項 5 または請求項 6 のいずれかに記載の用紙排出装置であって、

上記スプールデータ参照手段は上記格納されているデータを用いて、分割された上記所定態様のデータを復元可能であることを特徴とする用紙排出装置。

【請求項 8】

上記請求項 5 ~ 請求項 7 のいずれかに記載の用紙排出装置において、

上記データ検出手段は、印刷ジョブの終了を示すデータを検出することを特徴とする用紙排出装置。

30

【請求項 9】

印刷ジョブがスプーラによってキャンセルされたときに印刷装置にて印刷用紙を排出させる用紙排出方法であって、

上記スプーラが出力するスプールデータ中の後尾から所定サイズのデータを格納するスプールデータ参照工程と、

上記スプーラによるキャンセル時に起動され、上記スプールデータ参照工程により格納されているデータを参照して所定態様のデータの有無を検出するデータ検出工程と、

同データ検出工程によって所定態様のデータが検出されなかったときに上記印刷装置に対して用紙排出コマンドを出力する用紙排出コマンド出力工程とを含み、

上記所定サイズは、上記スプーラが出力する最大のデータサイズよりも小さいことを特徴とする用紙排出方法。

40

【請求項 10】

上記請求項 9 に記載の用紙排出方法であって、

上記スプールデータ参照工程は、上記所定態様データの検出に必要なビット数のデータを参照することを特徴とする用紙排出方法。

【請求項 11】

上記請求項 9 または請求項 10 のいずれかに記載の用紙排出方法であって、

上記スプールデータ参照工程は上記格納されているデータを用いて、分割された上記所定態様のデータを復元可能であることを特徴とする用紙排出方法。

50

【請求項 12】

上記請求項9～請求項11のいずれかに記載の用紙排出方法において、

上記データ検出工程は、印刷ジョブの終了を示すデータを検出することを特徴とする用紙排出方法。

【発明の詳細な説明】**【0001】****【発明の属する技術分野】**

本発明は、用紙排出プログラムを記録した媒体、データ識別プログラムを記録した媒体、用紙排出装置、データ識別装置、用紙排出方法およびデータ識別方法に関する。

【0002】

10

【従来の技術】

コンピュータから文書や図面等をプリンタにて印刷するためには、通常コンピュータ上でドライバ、スプーラおよびランゲージモニタ等が連携して印刷データをプリンタに転送している。すなわち、ドライバが生成する印刷データをスプーラが一時的にスプールし、ランゲージモニタがプリンタの状態を監視するなどして適正量のデータを適正タイミングでプリンタに供給する。

【0003】**【発明が解決しようとする課題】**

上述した従来の印刷実行においては、以下の問題があった。すなわち、利用者が印刷を取り消すときに、ドライバにて印刷を取り消せば当該ドライバがプリンタにおいてキャンセル処理を行うためのコマンドを出力することができるものの、スプーラにデータが転送されてから利用者が印刷を取り消した場合には、プリンタが適正なキャンセル処理を行うことができず印刷用紙が排出されない場合があった。かかる場合にはプリンタ自体に備える排出ボタンにて印刷用紙を排出することなども考えられるが、当該排出ボタンを設けないような低コストのプリンタではかかる排出は不可能であり、電源を切ってからプリンタのカバーを開けて印刷用紙を取り出す必要がある。

20

【0004】

本発明は、上記課題にかんがみてなされたもので、スプーラによる印刷キャンセルがなされたときも利用者の手を煩わすことなくプリンタに印刷用紙を排出させることができ可能な用紙排出プログラムを記録した媒体、データ識別プログラムを記録した媒体、用紙排出装置、データ識別装置、用紙排出方法およびデータ識別方法の提供を目的とする。

30

【0005】**【課題を解決するための手段】**

上記目的を達成するため、請求項1にかかる発明は、印刷ジョブがスプーラによってキャンセルされたときに印刷装置にて印刷用紙を排出させる用紙排出プログラムを記録した媒体であって、上記スプーラが出力するスプールデータ中の後尾から所定サイズのデータを格納するスプールデータ参照機能と、上記スプーラによるキャンセル時に起動され、上記スプールデータ参照機能により格納されているデータを参照して所定態様のデータの有無を検出するデータ検出機能と、同データ検出機能によって所定態様のデータが検出されなかったときに上記印刷装置に対して用紙排出コマンドを出力する用紙排出コマンド出力機能とをコンピュータに実現させ、上記所定サイズは、上記スプーラが出力する最大のデータサイズよりも小さい構成としてある。

40

【0006】

上記のように構成した請求項1にかかる発明においては、上記データ検出機能がスプーラによる印刷ジョブのキャンセル時に起動されるようになっており、当該キャンセル時にスプールデータ中の後尾データに所定態様のデータがあるか否かを検出する。そして、所定態様のデータが検出されなかったときに上記印刷装置に対して用紙排出コマンドを出力する。すなわち、印刷ジョブを終了させるためには通常ドライバがその終了を示す何らかのデータをスプールデータの最後に所定態様で付し、正常に印刷を終了するとかかる所定態様データによって印刷装置において適正に用紙が排出される。

50

【0007】

しかし、スプーラにおいて印刷ジョブをキャンセルしたときには、途中でスプールデータの送信が停止し、上記所定様のデータは印刷装置に送信されない。そこで、データ検出機能によって上記所定様データが検出されないときに用紙排出コマンドを出力する。従って、このようなスプーラによるキャンセル時にも印刷装置において用紙が排出されずに停止したままになることはない。尚、このデータ検出機能は上記スプーラによるキャンセル時に起動されるので、スプーラが処理を実行中にたまたま処理が滞ってしばらくデータが送信されなかった場合に用紙排出コマンドを出力してしまうようなことはない。

【0008】

また、上述のようにスプーラによるキャンセル時にデータ検出機能が実行されるように構成するためには様々な様が考えられ、例えば、ランゲージモニタ等のモジュールとして構成され、スプーラによるキャンセル時に実行される End Document 関数の処理の一部として実現することなどが考えられる。

10

【0009】

このように、データ検出機能にて所定様のデータの有無を検出するためには、上記スプールデータ参照機能においてスプールデータの後尾から所定サイズのデータを参照することができればよい。その構成としては、例えば、上記所定様データの検出に必要なビット数のデータを参照する構成とすることができます。この場合、さらには参照する後尾から所定サイズのデータを格納可能に構成すると好適である。

【0010】

20

すなわち、上記スプーラは通常大きな印刷データを処理する際には当該印刷データをいくつかのファイルに分割して印刷装置に対して出力する。この場合、偶然一番最後の出力データが上記所定様データのビット数に満たないものであることもあり得る。そこで、分割して出力されるデータの後尾から所定サイズのデータを格納しておけば、当該格納データと一緒に一番最後の出力データとを使用して所定様データのビット数分のデータを参照することができる。ここで所定サイズとは、上記スプーラが分割して出力するファイルの最大のデータサイズよりも小さいサイズである。さらに、このように参照する所定様データは種々のものが考えられるが、その好適な一例として、通常ドライバによってスプールデータの最後尾に付与される印刷ジョブの終了を示すデータを使用することができる。

30

【0011】

また、このように転送データ中において所定様のデータの有無を検出し、当該検出結果に応じて何らかの処理を実行する手法は、上述の印刷装置における排紙処理に限られず種々のものにて採用可能である。そこで、中間転送データ中に含まれうる所定様のデータを識別するためのデータ識別プログラムを記録した媒体であって、中間転送データを取得する中間転送データ取得機能と、同中間転送データ取得機能が取得した中間転送データ中の所定位置を参照する中間転送データ参照機能と、同中間転送データ参照機能が参照する所定位置のデータが所定様のデータであるか否かを判別するデータ様判別機能とをコンピュータに実現させる構成も考えられる。

【0012】

40

上記のように構成すると、中間転送データ中に含まれうる所定様のデータを識別する。すなわち、中間転送データを取得して当該所定データが記載されうる所定位置を参照し、当該位置に所定様のデータがあるか否かを判別する。この結果、転送データをすべてスキャンすることなく所定様のデータを発見することが可能になり、当該所定様のデータに基づいて何らかの処理を実行する場合の処理が非常に簡単になり、その処理も高速になる。

【0013】

むろん、以上述べてきた記録媒体は、磁気記録媒体であってもよいし光磁気記録媒体であってもよいし、今後開発されるいかなる記録媒体においても全く同様に考えることができる。また、一次複製品、二次複製品などの複製段階については全く問う余地無く同等で

50

ある。上記媒体とは異なるが、供給方法として通信回線を利用して行なう場合であれば通信回線が伝送媒体となって本発明が利用されることになる。

さらに、一部がソフトウェアであって、一部がハードウェアで実現されている場合においても発明の思想において全く異なるものではなく、一部を記録媒体上に記憶しておいて必要に応じて適宜読み込まれるような形態のものとしてあってもよい。

【0014】

このように、スプーラが出力するスプールデータ中の後尾データを参照して所定態様のデータの有無を検出する手法は実体のあるコンピュータにおいて実現され、その意味で本発明をそのような装置としても適用可能であることは容易に理解できる。このため、請求項5～請求項8にかかる発明においても、基本的には同様の作用となる。むろん、このような装置は単独で実施される場合もあるし、ある機器に組み込まれた状態で他の方法とともに実施されることもあるなど、発明の思想としてはこれに限らず、各種の態様を含むものであって、適宜、変更可能である。

【0015】

また、このような用紙排出プログラムは、かかる制御に従って処理を進めていく上でその根底にはその手順に発明が存在するということは当然であり、方法としても適用可能であることは容易に理解できる。このため、請求項9～請求項12にかかる発明においても、基本的には同様の作用となる。すなわち、必ずしも実体のある媒体などに限らず、その方法としても有効であることに相違はない。

【0016】

【発明の効果】

以上説明したように請求項1、請求項5、請求項9にかかる発明によれば、印刷ジョブがスプーラによってキャンセルされた場合でも的確に印刷用紙を排出させることができる。

また、請求項2、請求項6、請求項10にかかる発明によれば、簡単に所定態様のデータの有無を検出することができる。

さらに、請求項3、請求項7、請求項11にかかる発明によれば、確実に所定態様のデータの有無を検出することができる。

さらに、請求項4、請求項8、請求項12にかかる発明によれば、簡易な構成によりスプーラによるキャンセル時の処理を行うことができる。

【0017】

【発明の実施の形態】

以下、図面にもとづいて本発明の実施形態を説明する。

図1はコンピュータに周辺機器として印刷装置が接続されている状態の概略ハードウェア構成を示しており、図2は本発明がコンピュータにおいて複数のモジュールからなる用紙排出プログラムとして実現された場合における概略構成図を示している。

【0018】

まず、図1に示す概略ハードウェア構成について説明する。コンピュータ10は演算処理の中核をなすCPU11を備えており、このCPU11はシステムバス12を介してBIOSなどの記載されたROM13やRAM14にアクセス可能となっている。また、システムバス12には外部記憶装置としてのハードディスクドライブ15とフロッピーディスクドライブ16とCD-ROMドライブ17とが接続されており、ハードディスクドライブ15に記憶されたOS20やアプリケーションプログラム(APL)50がRAM14に転送され、CPU11はROM13とRAM14に適宜アクセスしてソフトウェアを実行する。すなわち、RAM14を一時的なワークエリアとして種々のプログラムを実行する。

【0019】

シリアル通信用I/O19aにはキーボード41やマウス42の操作用入力機器が接続され、図示しないビデオボードを介して表示用のディスプレイ18も接続されている。さらに、印刷装置30とはパラレル通信用I/O19bを介して接続可能となっている。ま

10

20

30

40

50

た、本コンピュータ10の構成は簡略化して説明しているが、パーソナルコンピュータとして一般的な構成を有するものを採用することができる。もちろん、本発明が適用されるコンピュータはパーソナルコンピュータに限定されるものではない。この実施例はいわゆるデスクトップ型コンピュータであるが、ノート型であるとか、モバイル対応のものであっても良い。また、コンピュータ10と印刷装置30との接続インターフェースも上記パラレル通信用I/O19bに限る必要はなくUSB接続やSCSI接続等種々の接続態様を採用可能であるし、今後開発されるいかなる接続態様であっても同様である。

【0020】

この例では各プログラムの類はハードディスクドライブ15に記憶されているが、記録媒体はこれに限定されるものではない。例えば、フロッピーディスク16aであるとか、CD-ROM17aであってもよい。これらの記録媒体に記録されたプログラムはフロッピーディスクドライブ16やCD-ROMドライブ17を介してコンピュータ10にて読み込まれ、ハードディスクドライブ15にインストールされる。そして、ハードディスクドライブ15を介してRAM14上に読み込まれてコンピュータを制御することになる。

【0021】

また、記録媒体はこれに限らず、光磁気ディスクなどであってもよい。さらに、半導体デバイスとしてフラッシュカードなどの不揮発性メモリなどを利用することも可能であるし、モデムや通信回線を介して外部のファイルサーバにアクセスしてダウンロードする場合には通信回線が伝送媒体となって本発明が利用される。印刷装置30は図示しないCPU、ファームウェア等を備えており、当該ファームウェアに記載されたプログラムに従つて上記コンピュータ10から送信される印刷データを所定のI/Oを介して受信する。そして、印刷装置30では当該データに基づいて所定のモータでヘッドや印刷用紙搬送機構を駆動しつつ印刷を実行する。

【0022】

一方、本実施形態にかかるコンピュータ10では、プリントドライバ(PRTDRV)
21とスプーラ22とランゲージモニタ23とがOS20に組み込まれている。図2に示すように、APL50において利用者は上記キーボード41やマウス42を介して文書や画面等を作成し、印刷装置30にて印刷を実行するための指示を出すことが可能である。PRTDRV21は同APL50からの制御信号によってAPL50の扱うデータを印刷用データに変換する処理等を担い、印刷データはスプーラ22に対して出力される。また、PRTDRV21はこの印刷データの出力時にデータの最後尾に「n」ビットで印刷ジョブの終了を示すコマンドデータを付加するようになっている。

【0023】

スプーラ22はPRTDRV21から入力される印刷データを上記ハードディスクドライブ15にスプールするようになっており、さらに当該スプールデータは同スプーラ22によって所定の容量毎に分割され、各分割データ毎に上記ランゲージモニタ23に渡される。ランゲージモニタ23は上記パラレル通信用I/O19bを介して上記印刷装置30に対して受領分割データを出力するようになっている。すなわち、スプーラ22とランゲージモニタ23とが連携して、上記印刷装置30の印刷進捗状況に応じて印刷データを出力するようになっている。

【0024】

より具体的には、ランゲージモニタ23にはStartDocPort関数、WritePort関数23a、EndDocPort関数23b等種々の関数が備えられており、スプーラ22がその処理に応じて適切な関数を実行することによって印刷処理が進行する。印刷開始時にはOpenPort関数とStartDocPort関数とが実行され、上記PRTDRV21の処理対象である印刷装置30がその装置名やポート名で指定されるなどして、上記パラレル通信用I/O19bを介する双方向通信可能状態が確保される。

【0025】

スプーラ22がランゲージモニタ23に対して上記分割データを出力する際にはWritePort関数23aを実行する。

10

20

30

40

50

tePort 関数 23a が実行されるようになっている。例えば、一回の印刷ジョブにかかる印刷データを 4 つに分割して出力する際には、当該 WritePort 関数 23a が 4 回実行され、ランゲージモニタ 23 はかかる分割データ毎にパラレル通信用 I/O 19b を介して印刷装置 30 に対してデータを送信する。また、本実施形態にかかる WritePort 関数 23a においては上記各分割データの後尾の「n」ビットのデータをセーブデータとして上記 RAM 14 に格納するようになっている。尚、分割して入力されるデータが「n」ビットに満たないときは、後述するように印刷データ全体として後ろから「n」ビットのデータをセーブデータとして確保する。

【0026】

印刷ジョブの終了時およびスプーラ 22 における印刷ジョブのキャンセル時には上記ランゲージモニタ 23 においては EndDocPort 関数 23b と ClosePort 関数とが実行され、上記パラレル通信用 I/O 19b の解放等の処理がなされる。EndDocPort 関数 23b においては上記セーブデータが上記印刷ジョブの終了を示すコマンドデータと一致するか否かを判別するようになっており、両者が一致しないときにパラレル通信用 I/O 19b を介して印刷用紙排出のためのリセットコマンドを出力させる。

【0027】

すなわち、EndDocPort 関数 23b がスプーラ 22 によるキャンセル時に起動された場合には、スプーラ 22 はデータの送信を途中で中断するので印刷ジョブの終了を示すコマンドデータがランゲージモニタ 23 に入力されず、当該コマンドを印刷装置 30 に出力することができない。この場合は、上記セーブデータが上記印刷ジョブの終了を示すコマンドデータと一致しないので、リセットコマンドを出力させて印刷用紙を排出させる。一方、EndDocPort 関数 23b が印刷ジョブの終了時に起動された場合には、上記セーブデータが上記印刷ジョブの終了を示すコマンドデータと一致するので、リセットコマンドを出力させることはない。

【0028】

図 3 はかかる構成において上記 WritePort 関数 23a が実行する上記セーブデータにかかる処理のフローを示している。ここで、上記分割データはスプーラ 22 によって所定の容量に分けられるものであり、最後に入力される分割データの前までは一つの印刷データについて等容量である。しかし、印刷データは常に等容量に分割できるわけではないので、等分割できない部分は最後の分割データで調整するようになっている。また、スプーラ 22 によって途中で印刷ジョブがキャンセルされたときにはデータの送信は途中で中断される。従って、最後に入力されるデータあるいはスプーラ 22 がキャンセルした時点でのデータに関してはそれ以前の分割データと等容量になるとは限らない。

【0029】

そこで、以降はいずれの場合であっても上記 WritePort 関数 23a に実際に入力される各データをバッファデータと呼ぶことにし、そのサイズを「m」ビットとする。同図の説明において同バッファデータの後ろに「x」と付し、バッファデータが最初から何番目のビットのものかを示すものとしてある。従って、バッファデータ「x」という表記において当該「x」は「1 ~ m」の整数である。また、上記セーブデータの後ろにも同様に「x」と付してあり、当該セーブデータが最初から何番目のものかを示し、セーブデータ「x」という表記において当該「x」は「1 ~ n」の整数である。

【0030】

本 WritePort 関数 23a は、上記バッファデータとそのサイズ「m」とをパラメタとして付与されつつ起動されるようになっており、図 3 に示すフローにおいては、分割データが上述の「n」ビットに満たない場合も含めて後に EndDocPort 関数 23b で適正な判別が行えるようにセーブデータを確保するための処理を行っている。このためにまず、ステップ S100 では上記分割データサイズ「m」が上記印刷ジョブの終了を示すコマンドサイズ「n」より小さいか否かを判別する。

【0031】

同ステップ S100 にてバッファデータサイズ「m」が上記印刷ジョブの終了を示すコ

10

20

30

40

50

マンドサイズ「n」より小さいと判別されないときには、ステップS110～S116の処理によって上記バッファデータの後尾「n」ビットのデータをセーブデータとして上記RAM14に格納する。具体的には、ステップS110にて当該処理に使用するカウンタ「i」に「1」を代入して初期化し、ステップS112にてセーブデータ[i]にバッファデータ[m-n+i]を代入する。

【0032】

ここで、バッファデータの「m-n+1」番目というのは当該バッファデータの後尾「n」ビットの先頭である。従って、上記ステップS112の処理によってセーブデータ[i]の先頭がバッファデータの後尾「n」ビットの先頭になる。

【0033】

ステップS114ではカウンタ「i」をインクリメントし、ステップS116にてカウンタ「i」が終了コマンドサイズ「n」より大きいか否かを判別する。そして、同ステップS116にてカウンタ「i」が終了コマンドサイズ「n」より大きいと判別されるまでステップS112以降の処理を繰り返す。すなわち、セーブデータ[1]～[n]までのすべてに対し、バッファデータの後尾データを格納する。

【0034】

上記ステップS100にてバッファデータサイズ「m」が上記印刷ジョブの終了を示すコマンドサイズ「n」より小さいと判別された時には、ステップS120～S136の処理によって、前バッファデータの後尾と現バッファデータとを併せて正味「n」ビット分のセーブデータを確保する。ここで、前バッファデータとは、上記各分割データの処理を実行するためにWritePort関数23aが複数回起動されるが、現在起動されているWritePort関数23aの直前に起動されていたものにおけるバッファデータを意味している。

【0035】

具体的には、ステップS120にて当該処理に使用するカウンタ「i」に「1」を代入して初期化し、ステップS122にてセーブデータ[i]に前セーブデータ[m+i]を代入する。前回のWritePort関数23aによって格納された前セーブデータ[i]は「n」ビットである。従って、セーブデータの先頭は前セーブデータの「m+1」番目である。

【0036】

上記ステップS122における処理の後ステップS124にてカウンタ「i」をインクリメントし、ステップS126にて同カウンタ「i」が「n-m」より大きいか否かを判別する。そして、同ステップS126にてカウンタ「i」が「n-m」より大きいと判別されるまで上記ステップS122～ステップS126までの処理を繰り返す。この結果、上記前セーブデータ[m+1]～[n]までがセーブデータ[1]～[n-m]までとして新たに確保される。

【0037】

さらに、ステップS130にて当該処理に使用するカウンタ「i」に再び「1」を代入して初期化し、ステップS132にてセーブデータ[n-m+i]に現バッファデータ[i]を代入する。すなわち、上述のステップS122～ステップS126にて生成したセーブデータ[1]～[n-m]の次に現バッファデータの先頭を対応させて代入する。ステップS134ではカウンタ「i」をインクリメントし、ステップS136にて同カウンタ「i」が「m」より大きいか否かを判別する。

【0038】

そして、同ステップS136にてカウンタ「i」が「m」より大きいと判別されるまで上記ステップS132～ステップS136までの処理を繰り返す。この結果、上記バッファデータ[1]～[m]までがセーブデータ[n-m+1]～[n]までとして新たに確保される。すなわち、上述の処理によって前セーブデータの後尾の必要ビット数分と現バッファデータとをあわせたものが現セーブデータとして確保され、結果としてセーブデータが「n」ビットになる。

10

20

30

40

50

【0039】

図4はかかる構成において上記EndDocPort関数23bが実行する上記セーブデータと終了コマンドとの比較にかかる処理のフローを示している。ステップS200では、上記RAM14に格納されているセーブデータを参照し、当該ランゲージモニタがデフォルトで有している上記終了コマンドデータと上記参照データとを比較する。同ステップS200にて両者が一致すると判別されたときには、終了コマンドデータ等を付すなどの処理を特に実行することではなく、通常のEndDocPort関数23bの機能を実行する。

【0040】

上記ステップS200にて両者が一致すると判別されないときには、ステップS210にてパラレル通信用I/O19bを介して印刷用紙排出のためのリセットコマンドを出力させる。本EndDocPort関数23bが起動されるのは上述のように印刷ジョブの終了時およびスプーラ22における印刷ジョブのキャンセル時である。印刷ジョブの終了時には上述のWritePort関数23aの処理によってセーブデータが終了コマンドになっているので、EndDocPort関数23bにて何ら処理を行わなくても当該終了コマンドが上記印刷装置30に出力される。

【0041】

一方、スプーラ22における印刷ジョブのキャンセル時には印刷データが最後まで出力されず、上述のWritePort関数23aの処理によって取得したセーブデータは終了コマンドになっている。しかし、ステップS210の処理によってリセットコマンドが印刷装置30に対して出力されるので、印刷装置30において印刷用紙の途中まで印字した段階で適切に当該印刷用紙が排出される。

【0042】

次に上記構成およびフローによって実現される印刷ジョブ処理の具体例を説明する。ここでは、スプーラ22にて印刷ジョブのキャンセルを実行した場合の例を示す。PRTD RV21は上記APL50にて印刷指示等がなされたときに、上記所定の処理を実行して当該処理データをひとまとまりの状態でスプーラ22に出力する。このとき、PRTD RV21は印刷データの後尾に終了コマンドを付している。

【0043】

同印刷データが与えられると同スプーラ22はこの印刷データをスプールするとともに所定の容量毎に分割し、これらを印刷データ列の順番通りに出力していく。

【0044】

ランゲージモニタ23においては、スプーラ22が分割データを出力するたびに上記WritePort関数23aが実行される。本具体例においては、2番目の分割データの処理においてバッファデータサイズは終了コマンドサイズより大きいので上記ステップS100の判別によりバッファデータの後尾「n」ビットをセーブデータ[i]とする。尚、ランゲージモニタ23が取得した1番目と2番目の分割データは上記パラレル通信用I/O19bに対して出力されるので、上記印刷装置30においては、これら1番目と2番目の分割データに基づく印刷は実行される。

【0045】

上述のようにスプーラ22にて印刷ジョブのキャンセルが実行された場合には、ランゲージモニタ23にて上記EndDocPort関数23bが起動される。ここで、上記セーブデータは上記終了コマンドではないのでステップS200の判別を経てステップS210において上記リセットコマンドを出力させる。従って、印刷装置30においては上記1番目と2番目の分割データに基づく印刷が実行された後に印刷用紙が排出され、スプーラ22におけるキャンセル時において印刷装置30の印刷が途中で停止して印刷用紙が中途半端に送られたままになることはない。

【0046】

さらに、他の具体例として、スプーラ22によってキャンセルが実行されることなく最後まで印刷ジョブを実行し、かつ、分割データの最後が終了コマンドサイズに満たない場

10

20

20

30

40

50

合を説明する。この場合、上述の例と同様に P R T D R V 2 1 が上記 A P L 5 0 にて印刷指示等がなされたときに上記所定の処理を実行し、当該処理データの後尾に終了コマンドを付しつつひとまとまりの状態でスプーラ 2 2 に出力する。

【0047】

同印刷データが与えられると同スプーラ 2 2 はこの印刷データをスプールするとともに所定の容量毎に分割し、これらを印刷データ列の順番通りに出力していく。ランゲージモニタ 2 3 においては、スプーラ 2 2 が分割データを出力するたびに上記 W r i t e P o r t 関数 2 3 a が実行される。スプーラ 2 2 が入力された印刷データを 4 つに分割し、4 番目の分割データが終了コマンドサイズに満たないような具体例においては、4 番目の分割データの処理においてバッファデータサイズは終了コマンドサイズより小さいので上記ステップ S 1 0 0 の判別により 3 番目の分割データのセーブデータを使用して、ステップ S 1 2 0 ~ S 1 2 6 の処理において必要なビット数のデータをセーブデータ [i] の先頭から順に格納し、ステップ S 1 3 0 ~ S 1 3 6 の処理において 4 番目のバッファデータを加えてセーブデータ [i] とする。

10

【0048】

本具体例においては、スプーラ 2 2 においてキャンセルが実行されることなく印刷ジョブの処理が進行するので、ランゲージモニタ 2 3 は分割データの 1 番目から 4 番目までのすべてを上記パラレル通信用 I / O 1 9 b に出力する。従って、印刷装置 3 0 では印刷用紙の排出処理を含めて適正に印刷を実行する。また、スプーラ 2 2 が 4 番目の分割データまでのすべてを出力すると、ランゲージモニタ 2 3 においては E n d D o c P o r t 関数 2 3 b が実行されるが、同 E n d D o c P o r t 関数 2 3 b のステップ S 2 0 0 ではセーブデータが終了コマンドと一致すると判別するので、上記リセットコマンドを出力することはない。

20

【0049】

このように、本発明においてはスプーラによるキャンセル時にスプールデータ中の後尾データを参照し、当該参照後尾データに所定態様のデータが検出されなかったときに、印刷用紙に対して用紙排出コマンドを出力する。従って、スプーラによる印刷キャンセルがなされたときも利用者の手を煩わすことなくプリンタに印刷用紙を排出させることができる。

30

【図面の簡単な説明】

【図 1】 コンピュータに周辺機器として印刷装置が接続されている状態の概略ハードウェア構成の概略図である。

30

【図 2】 用紙排出プログラムのモジュールの概略構成図である。

【図 3】 W r i t e P o r t 関数が実行するセーブデータにかかる処理のフローチャートである。

【図 4】 E n d D o c P o r t 関数が実行するセーブデータと終了コマンドとの比較にかかる処理のフローチャートである。

【符号の説明】

1 0 ... コンピュータ

40

1 1 ... C P U

1 2 ... システムバス

1 3 ... R O M

1 4 ... R A M

1 5 ... ハードディスクドライブ

1 6 ... フロッピーディスクドライブ

1 6 a ... フロッピーディスク

1 7 ... C D - R O M ドライブ

1 7 a ... C D - R O M

1 8 ... ディスプレイ

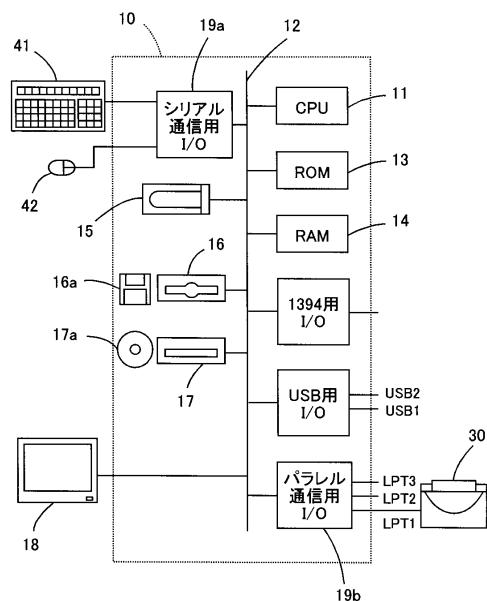
1 9 a ... シリアル通信用 I / O

50

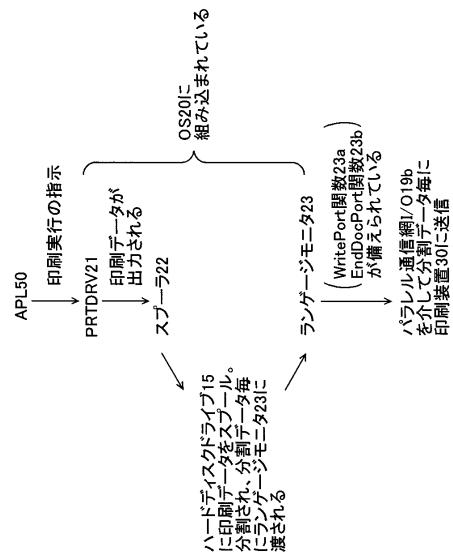
1 9 b ... パラレル通信用 I / O
 2 0 ... O S
 2 1 ... P R T D R V
 2 2 ... スプーラ
 2 3 ... ランゲージモニタ
 2 3 a ... W r i t e P o r t 関数
 2 3 b ... E n d D o c P o r t 関数
 3 0 ... 印刷装置
 4 1 ... キーボード
 4 2 ... マウス
 5 0 ... A P L

10

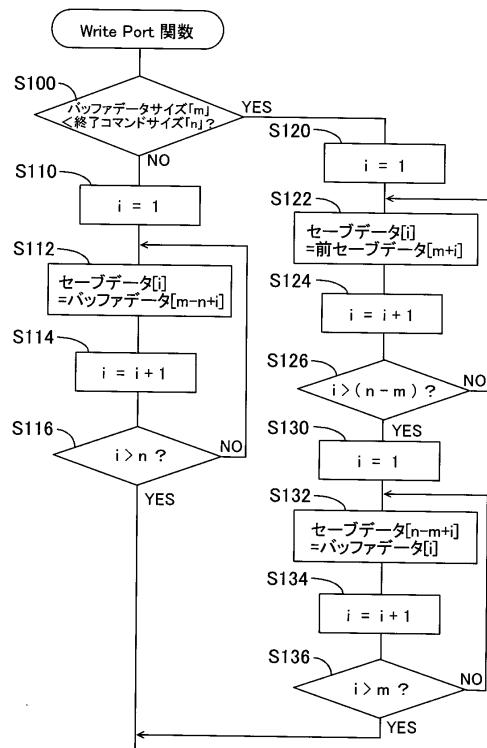
【図1】



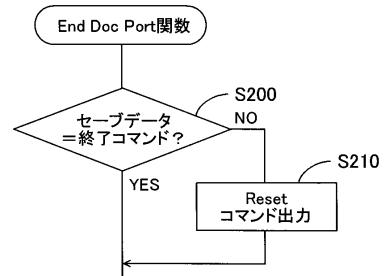
【図2】



【図3】



【図4】



フロントページの続き

(56)参考文献 特開平11-134120(JP,A)
特開2000-259368(JP,A)

(58)調査した分野(Int.Cl., DB名)

G06F 3/12

B41J 29/38

B65H 7/06

B65H 43/04