

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6164823号
(P6164823)

(45) 発行日 平成29年7月19日 (2017. 7. 19)

(24) 登録日 平成29年6月30日 (2017. 6. 30)

(51) Int. Cl.	F I		
B 4 1 J 29/38	(2006. 01)	B 4 1 J	29/38 Z
B 4 1 J 29/46	(2006. 01)	B 4 1 J	29/38 B
G 0 6 F 3/12	(2006. 01)	B 4 1 J	29/46 Z
		G 0 6 F	3/12 3 1 0
		G 0 6 F	3/12 3 3 4

請求項の数 5 (全 12 頁)

(21) 出願番号	特願2012-263124 (P2012-263124)	(73) 特許権者	000001007
(22) 出願日	平成24年11月30日 (2012. 11. 30)		キヤノン株式会社
(65) 公開番号	特開2014-108545 (P2014-108545A)		東京都大田区下丸子3丁目30番2号
(43) 公開日	平成26年6月12日 (2014. 6. 12)	(74) 代理人	100126240
審査請求日	平成27年11月30日 (2015. 11. 30)		弁理士 阿部 琢磨
前置審査		(74) 代理人	100124442
			弁理士 黒岩 創吾
		(72) 発明者	金窪 幸男
			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
		審査官	岡▲崎▼ 輝雄

最終頁に続く

(54) 【発明の名称】 印刷装置

(57) 【特許請求の範囲】

【請求項 1】

印刷装置であって、
ネットワークを介してホスト装置と通信を実行するためのネットワークインタフェースと、

前記ホスト装置から受信した印刷データに基づいて印刷処理を実行する印刷手段と、
前記印刷装置の起動時に、前記ネットワークインタフェースの初期化と前記印刷手段の初期化とを実行する制御手段とを備え、

前記制御手段は、前記ネットワークインタフェースの初期化が完了したことに応じて、
前記印刷手段の初期化中に発生したエンジンエラーを前記ホスト装置に通知するための第1の通信ポートを開き、前記印刷手段の初期化が完了したことに応じて、前記印刷データを前記ホスト装置から受信するための第2の通信ポートを開き、

前記印刷手段の初期化中にエンジンエラーが発生した場合、当該エンジンエラーを解消するために、前記印刷装置は、電源OFFを伴う再起動を必要とすることを特徴とする印刷装置。

【請求項 2】

前記印刷手段の初期化中にエンジンエラーが発生した場合は、前記制御手段は、前記第1の通信ポートを使用して当該エンジンエラーを前記ホスト装置に通知することを特徴とする請求項1に記載の印刷装置。

【請求項 3】

前記印刷手段の初期化とは、前記印刷手段の機構が正常に動作するか否かをチェックする処理であることを特徴とする請求項 1 又は 2 に記載の印刷装置。

【請求項 4】

前記印刷手段の初期化には、定着器ヒーターが規定温度まで温調可能か否かをチェックする処理が少なくとも含まれることを特徴とする請求項 3 に記載の印刷装置。

【請求項 5】

前記印刷手段の初期化には、用紙搬送モータが正常に動作するか否かをチェックする処理が少なくとも含まれることを特徴とする請求項 3 又は 4 に記載の印刷装置。

【発明の詳細な説明】

【技術分野】

10

【0001】

本発明は、印刷装置、印刷装置の制御方法、及びプログラムに関する。

【背景技術】

【0002】

近年、印刷装置とホスト装置とを接続するインタフェースは、USB I/F や Ethernet ケーブルによるネットワーク I/F などが一般的である。これらのインタフェースは、印刷データをホスト装置から印刷装置へ送信するだけでなく、印刷装置で発生したエラー等のステータス情報を印刷装置からホスト装置へ送信するためにも使用される。

【0003】

印刷データやステータス情報の通信について、特許文献 1 には、印刷データの通信とステータス情報の通信とでそれぞれ異なる論理ポートを使用する構成が開示されている。印刷データの通信とステータス情報の通信とでそれぞれ異なる論理ポートを使用することで、それぞれの通信が競合せずに、印刷装置とホスト装置との間で効率的に通信を行うことができる。ここでいう論理ポートとは、物理的には 1 本の通信ケーブルで接続されるが、論理的に複数の通信ポートとして使用可能なポートのことである。例えば、USB I/F は複数の End Point を持つことが可能であり、それぞれの End Point が論理ポートに相当する。また、ネットワーク I/F は TCP/IP のポート番号を指定することで仮想的に複数の通信ポートを実現可能であり、この通信ポートが論理ポートに相当する。

20

【0004】

30

次に、一般的な印刷装置の電源 ON 時の初期化シーケンスを、図 1 を用いて説明する。印刷装置の電源が ON になると、まず OS (Operating System) が起動される (101)。そして OS の起動が完了すると、インタフェースの初期化 (102) とプリンタエンジンの初期化 (103) が並行して実行される。

【0005】

インタフェースの初期化とは、USB I/F やネットワーク I/F を通信可能な状態にすることである。ただし、インタフェースの初期化が完了したからといって、すぐに通信を実行できるわけではない。通信を実行するためには、通信ポートを開く必要がある。一方、プリンタエンジンの初期化とは、プリンタエンジンが印刷を実行できる状態にすることである。具体的には、例えば定着器ヒーターが規定温度まで温調可能かどうか、用紙搬送モータが正常に動作するかどうかなどの、各機構の動作チェックを行う。インタフェースの初期化とプリンタエンジンの初期化を比較すると、プリンタエンジンの初期化の方がインタフェースの初期化よりも時間がかかる。そして、インタフェースの初期化とプリンタエンジンの初期化が両方とも完了すると、印刷装置は印刷を実行可能な状態になる。

40

【0006】

次に、一般的な印刷装置について、印刷データの通信に使用する印刷データ通信ポートを開くタイミングと、エラー等のステータス情報の通信に使用するステータス情報通信ポートを開くタイミングについて説明する。図 2 は、一般的な印刷装置として第 1 タイプの印刷装置について説明する図である。第 1 タイプの印刷装置は、ホスト装置から送信された PDL (ページ記述言語) データを印刷装置が受信し、印刷装置が受信した PDL デー

50

タをビットマップデータに展開して印刷を実行する。第1タイプの印刷装置の特徴は、LCD等の表示部を備える点である。

【0007】

第1タイプの印刷装置の電源がONになると、まずOSが起動する(201)。そしてOSの起動が完了すると、インタフェースの初期化(202)とプリンタエンジンの初期化(203)が並行して実行される。第1タイプの印刷装置では、印刷データ通信ポートとステータス情報通信ポートは、両方ともプリンタエンジンの初期化が完了したタイミングで開かれる。

【0008】

第1タイプの印刷装置において、プリンタエンジンの初期化中にタイミング204でエンジンエラーが発生したとする。このとき、ステータス情報通信ポートは開かれていないが、表示部にエンジンエラーの内容を表示できるため、ステータス情報通信ポートを開くタイミングはプリンタエンジンの初期化が完了したタイミングで問題ないことがわかる。

10

【0009】

また、エンジンエラーが発生した場合は、ユーザがエンジンエラーを解消するための操作をした後に、印刷装置の電源OFFを伴う再起動が必要になる。そして印刷装置が再起動すると、印刷装置が既に保持している印刷データ(PDLデータ及び展開済みのビットマップデータ)が削除されてしまう。しかしながら第1タイプの印刷装置は、印刷データ通信ポートが開かれるタイミングはプリンタエンジンの初期化が完了したタイミングであるため、電源OFFを伴う再起動を実行したとしても、このとき印刷装置が印刷データを保持していることはない。従って、第1タイプの印刷装置については、印刷データ通信ポートを開くタイミングはプリンタエンジンの初期化が完了したタイミングで問題ないことがわかる。

20

【先行技術文献】

【特許文献】

【0010】

【特許文献1】特開2000-238363号公報

【発明の概要】

【発明が解決しようとする課題】

【0011】

30

ここで、上述した第1タイプの印刷装置とは異なる第2タイプの印刷装置について考える。第2タイプの印刷装置の特徴は、第1タイプの印刷装置とは異なり、LCD等の表示部を備えない点である。

【0012】

第2タイプの印刷装置について、印刷データ通信ポートを開くタイミングとステータス情報通信ポートを開くタイミングを第1タイプの印刷装置と同様にした場合に発生する課題を、図3を用いて説明する。第2タイプの印刷装置の電源がONになると、まずOSが起動する(301)。そしてOSの起動が完了すると、インタフェースの初期化(302)とプリンタエンジンの初期化(303)が並行して実行される。印刷データ通信ポートを開くタイミングとステータス情報通信ポートを開くタイミングは、図3では第1タイプの印刷装置と同様であり、両方ともプリンタエンジンの初期化が完了したタイミングで開かれる。

40

【0013】

プリンタエンジンの初期化中にタイミング304でエンジンエラーが発生すると、ユーザにエンジンエラーを通知する必要がある。しかしながら、第2タイプの印刷装置は表示部を備えておらず、またステータス情報通信ポートが開かれていないため、エラーをユーザに通知できないという課題がある。

【0014】

つまり、第2タイプの印刷装置について各通信ポートを開くタイミングを第1タイプの印刷装置と同様にしてしまうと、図3で説明した課題が発生してしまう。そこで本発明で

50

は、第2タイプの印刷装置の特徴を考慮し、適切なタイミングで印刷データ通信ポートとステータス情報通信ポートを開くことを目的とする。

【課題を解決するための手段】

【0015】

上述した課題を解決するために、本発明が提供する印刷装置は、印刷装置であって、ネットワークを介してホスト装置と通信を実行するためのネットワークインタフェースと、前記ホスト装置から受信した印刷データに基づいて印刷処理を実行する印刷手段と、前記印刷装置の起動時に、前記ネットワークインタフェースの初期化と前記印刷手段の初期化とを実行する制御手段とを備え、前記制御手段は、前記ネットワークインタフェースの初期化が完了したことに応じて、前記印刷手段の初期化中に発生したエンジンエラーを前記ホスト装置に通知するための第1の通信ポートを開き、前記印刷手段の初期化が完了したことに応じて、前記印刷データを前記ホスト装置から受信するための第2の通信ポートを開き、前記印刷手段の初期化中にエンジンエラーが発生した場合、当該エンジンエラーを解消するために、前記印刷装置は、電源OFFを伴う再起動を必要とすることを特徴とする。

10

【発明の効果】

【0017】

本発明によれば、プリンタエンジンの初期化中に発生したエラーをホスト装置に通知することができる。また、プリンタエンジンの初期化が完了したことに応じて印刷データ通信ポートを開くため、プリンタエンジンの初期化中は印刷装置が印刷データを受信しないようにできる。

20

【図面の簡単な説明】

【0018】

【図1】従来技術における、印刷装置の起動時の初期化処理を説明する図である。

【図2】従来技術における、第1タイプの印刷装置の初期化処理を説明する図である。

【図3】第2タイプの印刷装置における課題を説明する図である。

【図4】印刷システムの構成を示す図である。

【図5】実施形態1における、印刷装置400の初期化処理を示す図である。

【図6】実施形態1における、印刷装置400の初期化処理を示すフローチャートである。

30

【図7】実施形態2における、印刷装置400の初期化処理を示す図である。

【発明を実施するための形態】

【0019】

以下、本発明を実施するための最良の形態について図面を用いて説明する。なお、以下の実施の形態は特許請求の範囲に係る発明を限定するものでなく、また実施の形態で説明されている特徴の組み合わせの全てが発明の解決手段に必須のものとは限らない。

【0020】

(実施形態1)

図4を用いて本実施形態に係る印刷システムについて説明する。本実施形態の印刷システムは、印刷装置400とホスト装置420を備え、印刷装置400とホスト装置420はネットワーク410を介して互いに通信可能に接続されている。印刷装置400は、ホスト装置から送信されたPDLデータを受信する。そして印刷装置400は当該PDLデータをビットマップデータに展開し、展開したビットマップデータに基づいて用紙に印刷処理を実行する。

40

【0021】

ホスト装置420は例えばPCである。ユーザの操作に応じて、ホスト装置420は印刷データとしてPDLデータを印刷装置400に送信する。また、ホスト装置420には表示部が備えられていて、後述する印刷装置400から通知されたステータス情報を表示することができる。

【0022】

50

次に、印刷装置400の構成を説明する。CPU401は、ROM402に記憶された制御プログラムを読み出して、印刷装置400全体の動作を制御する。RAM403は、CPU401の主メモリ、ワークエリア等の一時記憶領域として用いられる。NVRAM404は不揮発性メモリであり、様々な情報を記憶する。HDD405は、フォントデータ、エミュレーションプログラム、フォームデータを記憶するための記憶領域として用いられる。

【0023】

なお、印刷装置400は、1つのCPU401が1つのメモリ(RAM403またはHDD405)を用いて後述する図6のフローチャートに示す各処理を実行するものとするが、他の態様であっても構わない。例えば、複数のCPUや複数のRAMまたはHDDを協働させて後述する図6のフローチャートに示す各処理を実行するようにすることもできる。

10

【0024】

プリンタ406はプリンタエンジンを備え、ホスト装置420から受信したPDLデータに基づいて、用紙に印刷処理を実行する。

【0025】

操作パネル407は、ユーザが印刷装置400を操作するためのスイッチや操作キー、LED表示器等を備える。

【0026】

ネットワークI/F409は、印刷装置400とホスト装置420との間で実行される通信を制御する。ネットワークI/F409には、EthernetケーブルやLANケーブル等のネットワークケーブルが接続される。

20

【0027】

印刷装置400はネットワークI/F409を介してホスト装置420から送信された印刷データ(PDLデータ)を受信したり、印刷装置400で発生したエラー等のステータス情報をホスト装置420に送信する。印刷データの通信とステータス情報の通信を1つの通信ポートで行うと、それぞれの通信が競合してしまい、通信が失敗する場合がある。そこで本実施形態では、印刷装置400は印刷データの通信とステータス情報の通信とでそれぞれ異なる論理ポートを使用する。ネットワークI/F409は、TCP/IPのポート番号を指定することで仮想的に複数の通信ポートを実現可能であり、この複数の通信ポートを論理ポートとしてそれぞれ印刷データの通信とステータス情報の通信とに割り当てる。

30

【0028】

印刷装置400とホスト装置420との間の通信は、ネットワークI/F409に限らない。印刷装置400はUSB I/F408を備え、USB I/F408にはUSBケーブルが接続される。そしてUSBケーブルを介して、印刷装置400とホスト装置420との間で通信を実行することもできる。USB I/Fは複数のEndpointを持つことが可能であり、それぞれのEndpointが、論理ポートとして印刷データの通信とステータス情報の通信とに割り当てられる。図4ではネットワークI/F409を介してホスト装置420と通信を行っているが、USB I/F408を用いて通信を行っても構

40

【0029】

以上、印刷装置400の構成を説明した。なお、印刷装置400は、印刷装置400で発生したエラー等のステータス情報を表示するための表示部を備えていない。仮に表示部を備えていたとしても、例えば表示領域が小さいという理由で、ステータス情報の詳細な表示はできないこととする。また、印刷装置400は、LED表示器やピープ音を用いて何らかのエラーが発生していることはユーザに通知できるかもしれないが、LED表示器やピープ音ではエラーの詳細な内容をユーザに通知することはできない。そこで印刷装置400は、印刷装置400で発生したエラー等のステータス情報をユーザに通知するために、インタフェース(USB I/F408又はネットワークI/F409)を介してホス

50

ト装置 4 2 0 にステータス情報を通知する。この通知を受けたホスト装置 4 2 0 は、ホスト装置 4 2 0 の表示部に通知されたステータス情報を表示する。

【 0 0 3 0 】

次に、印刷装置 4 0 0 の電源 ON 時、つまり印刷装置 4 0 0 の起動時に実行する初期化シーケンスを、図 5 を用いて説明する。印刷装置 4 0 0 の電源が ON になると、まず OS が起動する (5 0 1)。そして OS の起動が完了すると、インタフェースの初期化 (5 0 2) とプリンタエンジンの初期化 (5 0 3) が並行して実行される。

【 0 0 3 1 】

インタフェースの初期化とは、USB I / F 4 0 8 やネットワーク I / F 4 0 9 を通信可能な状態にすることである。つまり、インタフェースの初期化が完了すると、USB I / F 4 0 8 やネットワーク I / F 4 0 9 は通信可能な状態になる。ただし、インタフェースの初期化が完了したからといって、すぐに通信を実行できるわけではない。通信を実行するためには、通信ポートを開く必要がある。なお、通信ポートを開くとは、その通信ポートを使用した通信を許可すること、或いはホスト装置 4 2 0 との間でデータを送受信できる状態にすることである。また、通信ポートをオープンにする、通信ポートを開放するという表現も、本実施形態における「通信ポートを開く」と同義である。

10

【 0 0 3 2 】

一方、プリンタエンジンの初期化とは、プリンタ 4 0 6 が印刷を実行できる状態にすることである。つまり、プリンタエンジンの初期化が完了すると、プリンタ 4 0 6 は印刷を実行可能な状態になる。具体的には、例えば定着器ヒーターが規定温度まで温調可能かどうか、用紙搬送モータが正常に動作するかどうかなどの、各機構の動作チェックを行う。インタフェースの初期化とプリンタエンジンの初期化を比較すると、プリンタエンジンの初期化の方がインタフェースの初期化よりも時間がかかる。そして、インタフェースの初期化とプリンタエンジンの初期化が両方とも完了すると、印刷装置 4 0 0 は印刷を実行可能な状態になる。

20

【 0 0 3 3 】

次に、印刷データの通信に使用する印刷データ通信ポートを開くタイミングと、エラー等のステータス情報の通信に使用するステータス情報通信ポートを開くタイミングとについて説明する。印刷装置 4 0 0 は、5 0 2 で示すインタフェースの初期化が完了すると、ステータス情報通信ポートを開く。そして 5 0 3 で示すプリンタエンジンの初期化が完了すると、印刷データ通信ポートを開く。ステータス情報通信ポートを開くタイミングと印刷データ通信ポートを開くタイミングを図 5 で示すタイミングにするメリットを次に説明する。

30

【 0 0 3 4 】

印刷装置 4 0 0 において、プリンタエンジンの初期化中にタイミング 5 0 4 でエンジンエラーが発生したとする。印刷装置 4 0 0 は、エンジンエラーをユーザに通知するためには、エンジンエラーを示すステータス情報をホスト装置 4 2 0 に通知する必要がある。本実施形態では、プリンタエンジンの初期化の完了を待たずにインタフェースの初期化が完了したことに応じてステータス情報通信ポートが開かれているため、エンジンエラーをホスト装置 4 2 0 に通知することができる。もしステータス情報通信ポートを開くタイミングをプリンタエンジンの初期化が完了したタイミングにすると、図 3 で説明したように印刷装置 4 0 0 はエンジンエラーをユーザに通知できなくなってしまう。

40

【 0 0 3 5 】

プリンタエンジンの初期化中にエンジンエラーが発生した場合、ユーザがエンジンエラーを解消するための操作をした後に、印刷装置 4 0 0 の電源 OFF を伴う再起動が必要になる。印刷装置 4 0 0 は、電源 OFF を伴う再起動を行うと、印刷装置 4 0 0 が既に保持している印刷データを削除してしまう。そこで本実施形態では、印刷データ通信ポートを開くタイミングを、5 0 3 で示すプリンタエンジンの初期化が完了したタイミングとしている。もし印刷データ通信ポートを開くタイミングをインタフェースの初期化が完了したタイミングにすると、タイミング 5 0 4 の時点で既に印刷装置 4 0 0 が印刷データを保持

50

している場合がある。この場合、再起動によって印刷装置 400 が印刷データを削除してしまう。削除された印刷データを印刷装置 400 が再度取得しようとする、例えばホスト装置 420 においてユーザが再度印刷操作を行い、削除された印刷データと同じ印刷データを再度印刷装置に送信するという手間が発生してしまう。そこで本実施形態では、印刷データ通信ポートを開くタイミングを 503 で示すプリンタエンジンの初期化が完了したタイミングにすることで、プリンタエンジンの初期化中にホスト装置 420 から印刷データを受信しないようにすることができる。

【0036】

以上の説明の通り、印刷装置 400 は、502 で示すインタフェースの初期化が完了するとステータス情報通信ポートを開き、503 で示すプリンタエンジンの初期化が完了すると印刷データ通信ポートを開く。これにより、プリンタエンジンの初期化中に発生したエンジンエラーのユーザへの通知を実現しつつ、エンジンエラーの発生によって印刷データが削除されてしまうことを防止することができる。

10

【0037】

次に、印刷装置 400 の電源が ON になったとき、つまり印刷装置 400 の起動時に印刷装置 400 が実行する処理について、図 6 のフローチャートを用いて説明する。図 6 のフローチャートに示す各ステップは、CPU 401 が ROM 402 等のメモリに格納されたプログラムを RAM 403 に展開して実行することによって処理される。

【0038】

印刷装置 400 の電源が ON になると、ステップ S601 において、CPU 401 は OS を起動する。そしてステップ S602 において、CPU 401 は OS の起動が完了したか否かを判定する。OS の起動が完了したと CPU 401 が判定すると、ステップ S603 に進む。一方、OS の起動が完了していないと CPU 401 が判定すると、OS の起動が完了するまで待機する。

20

【0039】

次にステップ S603 において、CPU 401 はインタフェースの初期化とプリンタエンジンの初期化を開始する。インタフェースの初期化とは、印刷装置 400 が備えるインタフェース、即ち USB I/F 408 とネットワーク I/F 409 を通信可能な状態にすることである。一方、プリンタエンジンの初期化とは、プリンタ 406 が印刷を実行できる状態にすることである。なお、本実施形態では OS の起動が完了した後にインタフェースの初期化とプリンタエンジンの初期化を行っているが、OS の起動の完了を待たずにインタフェースの初期化とプリンタエンジンの初期化を行っても構わない。

30

【0040】

次にステップ S604 において、CPU 401 はインタフェースの初期化が完了したか否かを判定する。インタフェースの初期化が完了したと CPU 401 が判定すると、ステップ S605 に進む。一方、インタフェースの初期化が完了していないと CPU 401 が判定すると、インタフェースの初期化が完了するまで待機する。

【0041】

次にステップ S605 において、CPU 401 はステータス情報通信ポートを開く。USB I/F 408 の場合、複数の End Point の 1 つに割り当てられたステータス情報通信ポートを開く。また、ネットワーク I/F 409 の場合、TCP/IP のポート番号の 1 つに割り当てられたステータス情報通信ポートを開く。ステップ S605 でステータス情報通信ポートを開くことにより、印刷装置 400 は印刷装置 400 で発生したエラーをホスト装置 420 に通知できるようになる。

40

【0042】

次にステップ S606 において、CPU 401 はプリンタエンジンの初期化中にエンジンエラーが発生したか否かを判定する。エンジンエラーが発生したと CPU 401 が判定すると、ステップ S607 に進む。一方、エンジンエラーが派生していないと CPU 401 が判定すると、ステップ S608 に進む。なお、図 5 ではインタフェースの初期化が完了した後にエンジンエラーが発生した例を説明したが、インタフェースの初期化が完了す

50

る前にエンジンエラーが発生したとしても、ステップS606の処理によってエラーが発生したことをCPU401は検知できる。

【0043】

ステップS607において、CPU401はプリンタエンジンの初期化中にエンジンエラーをホスト装置420に通知する。本実施形態では、CPU401は、USB I/F408もしくはネットワークI/F409を介して接続されているホスト装置420に対して、ステップS605で開いたステータス情報通信ポートを使用してエンジンエラーをホスト装置420に通知する。エンジンエラーの通知を受けたホスト装置420は、エンジンエラーの内容をホスト装置420の表示部に表示する。ユーザはこれを確認することで、印刷装置400のプリンタエンジンの初期化中に発生したエンジンエラーを確認することができる。

10

【0044】

次にステップS608において、CPU401はプリンタエンジンの初期化が完了したか否かを判定する。プリンタエンジンの初期化が完了したとCPU401が判定すると、ステップS609に進む。一方、プリンタエンジンの初期化が完了していないとCPU401が判定すると、ステップS606に戻る。

【0045】

次にステップS609において、CPU401は印刷データ通信ポートを開く。USB I/F408の場合、複数のEndPointの1つに割り当てられた印刷データ通信ポートを開く。また、ネットワークI/F409の場合、TCP/IPのポート番号の1つに割り当てられた印刷データ通信ポートを開く。ステップS609で印刷データ通信ポートを開くことにより、印刷装置400はホスト装置から送信された印刷データを受信できるようになる。

20

【0046】

以上の説明の通り、本実施形態では、ステータス情報通信ポートを開くタイミングをインタフェースの初期化が完了したタイミングにする。これにより、プリンタエンジンの初期化中に発生したエンジンエラーをホスト装置420に通知できる。また、本実施形態では、印刷データ通信ポートを開くタイミングをプリンタエンジンの初期化が完了したタイミングにする。これにより、プリンタエンジンの初期化中、つまりプリンタエンジンのエラーが発生する可能性がある間は、ホスト装置420から印刷データを受信しないようにすることができる。

30

【0047】

なお、本実施形態ではインタフェースの例としてネットワークI/F（ネットワークインタフェース）とUSB I/F（USBインタフェース）を挙げて説明したが、他のインタフェースを使用して印刷装置400とホスト装置420との間で通信を実行してもよい。例えば、無線LANを用いて印刷装置400とホスト装置420との間で通信を実行してもよい。無線LANを用いる場合も、ステータス情報通信ポートと印刷データ通信ポートをそれぞれ異なる論理ポートに割り当て、ステータス情報通信ポートと印刷データ通信ポートを開くタイミングを図5で説明したタイミングにすればよい。

【0048】

（実施形態2）

実施形態1では、図5で説明したようにインタフェースの初期化が完了したことに応じてステータス情報通信ポートを開く構成を説明した。これに対して本実施形態では、実施形態1の変形例を図7を用いて説明する。

40

【0049】

本実施形態では、インタフェースの初期化が完了したことに応じてステータス情報通信ポートを開くのではなく、プリンタエンジンの初期化中にエンジンエラーが発生したことをCPU401が検知した場合に、CPU401がステータス情報通信ポートを開く。図7(a)では、タイミング701でエンジンエラーが発生し、CPU401がこのエンジンエラーを検知したことに応じてステータス情報通信ポートを開いている。そしてCPU

50

401は、ステータス情報通信ポートを使用してホスト装置420にエンジンエラーを通知する。

【0050】

プリンタエンジンの初期化中にエンジンエラーが発生しなかった場合は、図7(b)で示すように、プリンタエンジンの初期化が完了したことに応じてCPU401が印刷データ通信ポートとステータス情報通信ポートを両方開く。

【0051】

本実施形態によれば、エンジンエラーを検知したことに応じてステータス情報通信ポートを開くため、実施形態1と同様にエンジンエラーをホスト装置420に通知できる。また、本実施形態では、印刷データ通信ポートを開くタイミングをプリンタエンジンの初期化が完了したタイミングにする。これにより、実施形態1と同様に、プリンタエンジンの初期化中、つまりプリンタエンジンのエラーが発生する可能性がある間は、ホスト装置420から印刷データを受信しないようにすることができる。

10

【0052】

(その他の実施形態)

本発明は、以下の処理を実行することによっても実現される。即ち、上述した実施形態の機能を実現するソフトウェア(プログラム)を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ(またはCPUやMPU等)がプログラムを読み出して実行する処理である。

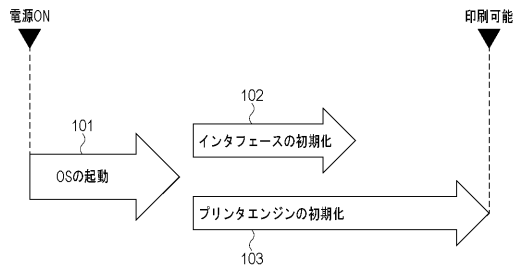
20

【符号の説明】

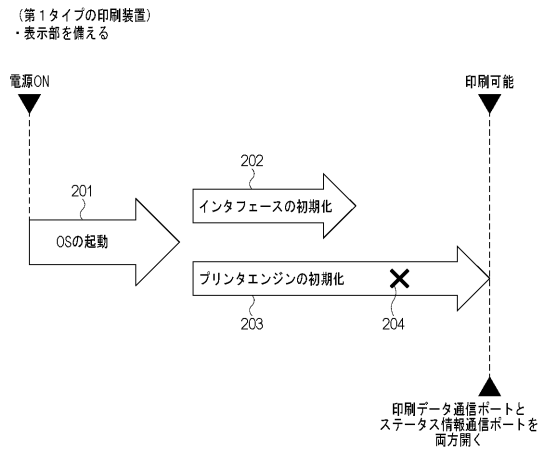
【0053】

- 400 印刷装置
- 401 CPU
- 408 USB I/F
- 409 ネットワーク I/F
- 420 ホスト装置

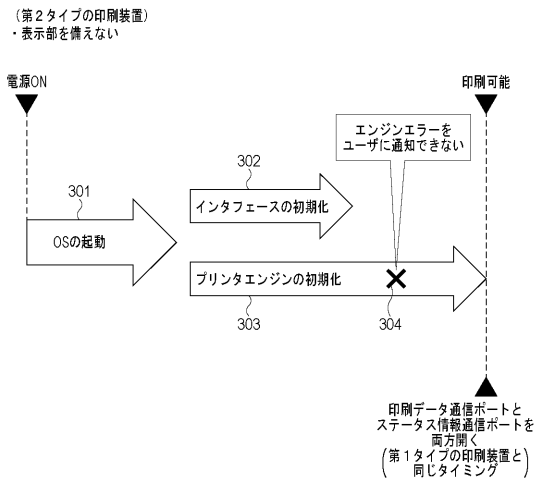
【図1】



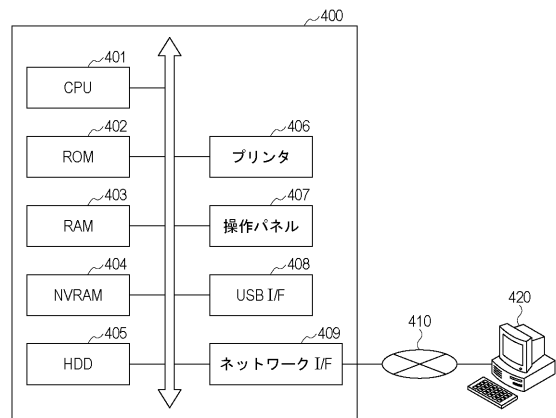
【図2】



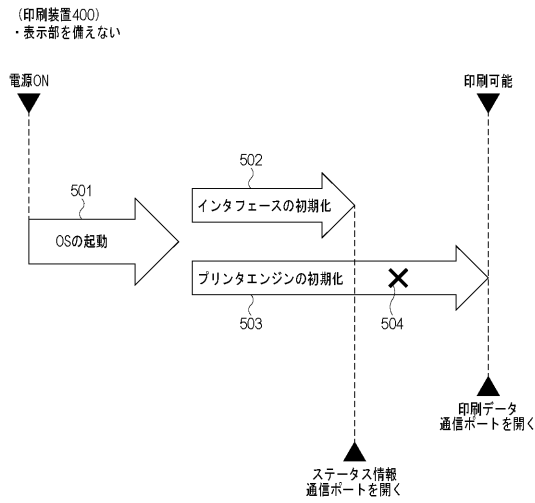
【図3】



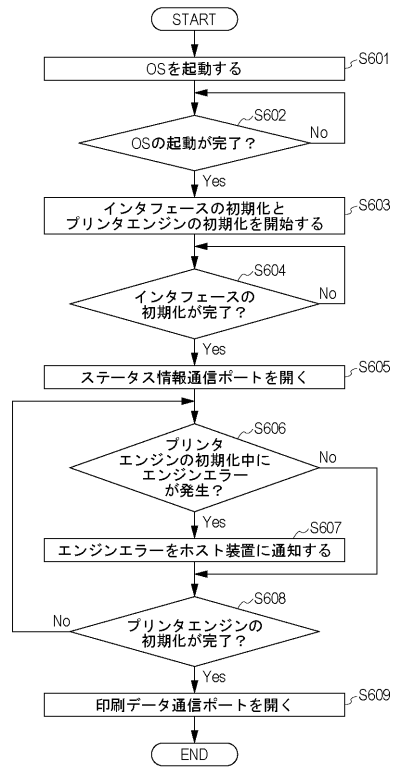
【図4】



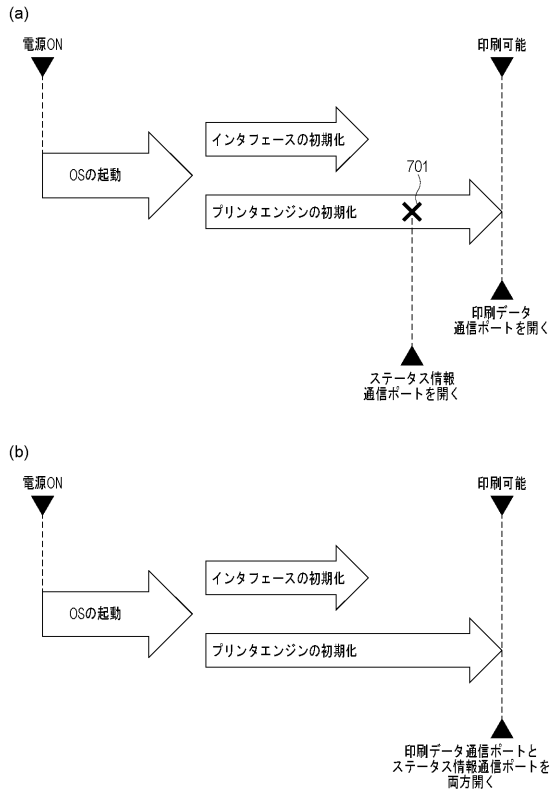
【図5】



【図6】



【図7】



フロントページの続き

- (56)参考文献 特開2003-316712(JP,A)
特開2003-067318(JP,A)
米国特許出願公開第2003/0197890(US,A1)
特開2004-221629(JP,A)
特開2002-055785(JP,A)
特開2009-194441(JP,A)
特開2000-135820(JP,A)

(58)調査した分野(Int.Cl., DB名)

B41J 29/38
B41J 29/46
G06F 3/12