

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第5868208号  
(P5868208)

(45) 発行日 平成28年2月24日 (2016. 2. 24)

(24) 登録日 平成28年1月15日 (2016. 1. 15)

(51) Int. Cl.

F I

**H04N 1/00 (2006.01)**

H04N 1/00 C

**B41J 29/38 (2006.01)**

B41J 29/38 D

**G03G 21/00 (2006.01)**

G03G 21/00 398

**G06F 3/12 (2006.01)**

G06F 3/12 321

請求項の数 11 (全 40 頁)

(21) 出願番号 特願2012-31392 (P2012-31392)  
 (22) 出願日 平成24年2月16日 (2012. 2. 16)  
 (65) 公開番号 特開2013-168827 (P2013-168827A)  
 (43) 公開日 平成25年8月29日 (2013. 8. 29)  
 審査請求日 平成27年2月5日 (2015. 2. 5)

(73) 特許権者 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  
 (74) 代理人 100145827  
 弁理士 水垣 親房  
 (74) 代理人 100199820  
 弁理士 西脇 博志  
 (72) 発明者 畑農 学海  
 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 橋爪 正樹

(56) 参考文献 特開2009-181467 (JP, A  
 )

最終頁に続く

(54) 【発明の名称】 画像形成装置、画像形成装置の制御方法、及びプログラム

(57) 【特許請求の範囲】

【請求項1】

画像形成装置であって、

指定期間内に前記画像形成装置をシャットダウンする電源断タスクを指定する第1指定手段と、

指定期間内に前記画像形成装置に設定されている設定の各項目に対する設定を行う設定工程と該設定工程の後に前記画像形成装置を再起動する再起動工程を行う設定タスクを指定する第2指定手段と、

前記指定された前記電源断タスクおよび前記設定タスクの実行を制御するタスク制御手段とを有し、

前記タスク制御手段は、設定タスクの設定工程における設定の開始前の待機中に電源断タスクによる前記画像形成装置のシャットダウンが開始される場合には、当該設定タスクを実行することなく当該電源断タスクによるシャットダウンが実行されるように制御し、

前記タスク制御手段は、さらに、設定タスクの再起動工程における再起動の開始前の待機中に電源断タスクによる前記画像形成装置のシャットダウンが開始される場合には、当該設定タスクの再起動工程における再起動を行うことなく当該電源断タスクによるシャットダウンが実行されるように制御することを特徴とする画像形成装置。

【請求項2】

前記タスク制御手段は、前記電源断タスクの指定期間の後端が、前記設定タスクの指定

期間の先端から設定工程が完了すると予想される日時までの期間に含まれる場合、前記設定タスクを一旦キャンセルするとともに前記画像形成装置の次回起動時に実行されるように再スケジューリングして、前記電源断タスクを実行するように制御することを特徴とする請求項 1 に記載の画像形成装置。

【請求項 3】

前記タスク制御手段は、前記電源断タスクの指定期間の後端が、前記設定タスクの設定工程が完了すると予想される日時から前記設定タスクの再起動工程により再起動される前記画像形成装置の再起動が完了すると予想される日時までの期間に含まれる場合、前記設定タスクの前記設定工程を実行し、さらに前記設定タスクの前記再起動工程をキャンセルし、前記電源断タスクを実行するように制御することを特徴とする請求項 1 又は 2 に記載の画像形成装置。

10

【請求項 4】

前記タスク制御手段は、前記電源断タスクの指定期間の後端が、前記設定タスクの再起動工程により再起動される前記画像形成装置の再起動が完了すると予想される日時より以降となる場合、前記設定タスクの前記設定工程を実行し、前記電源断タスクを一旦キャンセルするとともに前記画像形成装置の次回起動時に実行されるように再スケジューリングして、前記設定タスクの再起動工程を実行するように制御することを特徴とする請求項 1 乃至 3 のいずれか 1 項に記載の画像形成装置。

【請求項 5】

前記タスク制御手段が前記設定タスクの前記再起動工程をキャンセルしたにもかかわらず、前記電源断タスクが実行されなかった場合に、所定のユーザに通知を行う通知手段を有することを特徴とする請求項 3 に記載の画像形成装置。

20

【請求項 6】

前記通知は、手動により前記画像形成装置を再起動することを促す通知であることを特徴とする請求項 5 に記載の画像形成装置。

【請求項 7】

所定のユーザからの指示に応じて前記電源断タスクをキャンセルするキャンセル手段を有し、

前記電源断タスクが実行されなかった場合とは、前記キャンセル手段により前記電源断タスクがキャンセルされた場合を含むことを特徴とする請求項 5 又は 6 に記載の画像形成装置。

30

【請求項 8】

前記第 1 指定手段は、前記指定期間内に待機ジョブの実行が終了しない場合でも前記電源断タスクを実行する強制実行指示の有無を含む指示を行うものであり、

前記電源断タスクが実行されなかった場合とは、前記電源断タスクに前記強制実行指示がなされておらず前記電源断タスクの指定期間内に待機ジョブの実行が終了しなかった場合を含むことを特徴とする請求項 5 乃至 7 のいずれか 1 項に記載の画像形成装置。

【請求項 9】

前記タスク制御手段は、前記電源断タスクの指定期間の後端が、前記設定タスクの指定期間の先端から設定工程が完了すると予想される日時までの期間に含まれる場合であって、前記電源断タスクの指定期間の先端が、前記設定タスクの設定工程が完了すると予想される日時よりも先である場合には、前記電源断タスクを前記設定タスクの設定工程の直後に実行するように再スケジューリングして、前記設定タスクの設定工程を実行するように制御することを特徴とする請求項 1 に記載の画像形成装置。

40

【請求項 10】

画像形成装置における制御方法であって、

指定期間内に前記画像形成装置をシャットダウンする電源断タスクを指定する第 1 指定ステップと、

指定期間内に前記画像形成装置に設定されている設定の各項目に対する設定を行う設定工程と該設定工程の後に前記画像形成装置を再起動する再起動工程を行う設定タスクを指

50

定する第2指定ステップと、

前記指定された前記電源断タスクおよび前記設定タスクの実行を制御するタスク制御ステップとを有し、

前記タスク制御ステップでは、設定タスクの設定工程における設定の開始前の待機中に電源断タスクによる前記画像形成装置のシャットダウンが開始される場合には、当該設定タスクを実行することなく当該電源断タスクによるシャットダウンが実行され、

前記タスク制御ステップでは、さらに、設定タスクの再起動工程における再起動の開始前の待機中に電源断タスクによる前記画像形成装置のシャットダウンが開始される場合には、当該設定タスクの再起動工程における再起動を行うことなく当該電源断タスクによるシャットダウンが実行されることを特徴とする制御方法。

10

【請求項11】

コンピュータを、請求項1乃至9のいずれか1項に記載された手段として機能させるためのプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、画像形成装置、画像形成装置の制御方法、及びプログラムに関する。

【背景技術】

【0002】

従来、画像形成装置等のネットワークデバイス（以下、デバイス）の管理システムには、デバイスを電源断する機能および、デバイス設定を行う機能およびそれに伴って設定変更を反映させるためデバイスを再起動する機能を備えているものが存在する。

20

【0003】

また、それらを管理者からの指示により即時実行することや、スケジュール設定に従って自動的に実行することができるものも存在する。

また、それらの機能のスケジュール設定による自動実行が重なった場合に、少なくとも一方の時刻をずらして実行することでデバイス設定を行っている途中で電源断することを予め防止する機能を備えたものが存在する（特許文献1参照）。

【先行技術文献】

【特許文献】

30

【0004】

【特許文献1】特開2010-004335号公報

【発明の概要】

【発明が解決しようとする課題】

【0005】

しかしながら、それら従来のネットワークデバイス管理システムでは、それらの機能を管理者が指示して即時実行しようとした場合、電源断がデバイス設定を行っている途中に行われてしまう場合がある。この場合、電源断により、デバイス設定が正しく変更されることが妨げられてしまう、ことを防止できなかった。

【0006】

40

また、即時実行の指示であれ、スケジュール設定による自動実行であれ、再起動を実行しようとしている間に、電源断を行うべき時刻が過ぎてしまうと、電源断を正しく実行できなかった。

【0007】

このため、そのようなケースで電源断されているべき時刻に、電源が投入されたままの状態になってしまうことを防止できなかった。なお、外部からの電源供給が突然断たれてしまうといった事態は、デバイス故障の原因にもなりかねない。そのため、特に、停電やメンテナンス等によりデバイスに対する外部からの電源供給が予め特定の時刻に断たれることが分かっている場合には、その時刻までにデバイスを電源断することは重要であるが、従来の技術では、その時刻に電源断されないままになってしまうケースがあった。

50

## 【 0 0 0 8 】

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、画像形成装置の設定とそれに伴う再起動、および、電源断を、重なるようにそれぞれを即時実行またはスケジュール設定して指示された場合においても、正しく画像形成装置の設定と電源断を行うことを可能にする仕組みを提供することである。

## 【課題を解決するための手段】

## 【 0 0 0 9 】

本発明は、画像形成装置であって、指定期間内に前記画像形成装置をシャットダウンする電源断タスクを指定する第1指定手段と、指定期間内に前記画像形成装置に設定されている設定の各項目に対する設定を行う設定工程と該設定工程の後に前記画像形成装置を再起動する再起動工程を行う設定タスクを指定する第2指定手段と、前記指定された前記電源断タスクおよび前記設定タスクの実行を制御するタスク制御手段とを有し、前記タスク制御手段は、設定タスクの設定工程における設定の開始前の待機中に電源断タスクによる前記画像形成装置のシャットダウンが開始される場合には、当該設定タスクを実行することなく当該電源断タスクによるシャットダウンが実行されるように制御し、前記タスク制御手段は、さらに、設定タスクの再起動工程における再起動の開始前の待機中に電源断タスクによる前記画像形成装置のシャットダウンが開始される場合には、当該設定タスクの再起動工程における再起動を行うことなく当該電源断タスクによるシャットダウンが実行されるように制御することを特徴とする。

## 【発明の効果】

## 【 0 0 1 0 】

本発明によれば、画像形成装置の設定とそれに伴う再起動、および、電源断を、重なるようにそれぞれを即時実行またはスケジュール設定して指示された場合においても、正しく画像形成装置の設定と電源断を行うことが可能になる。

## 【 0 0 1 1 】

特に、電源断が確実に実行されているべき時刻において電源断が実行されずに終わることがないため、外部から電源供給を断たれる場合においても画像形成装置の故障を未然に防ぐことが可能になる。

## 【図面の簡単な説明】

## 【 0 0 1 2 】

【図1】本発明の一実施例を示す画像形成装置を適用可能なシステムの一例を示す図である。

【図2】MFP 102の構成の一部を模式化したブロック図である。

【図3】情報処理装置 103の構成を模式化したブロック図である。

【図4】MFP 102のコントローラ上 202上で動作する本発明の実施例1のネットワークデバイス管理システムのネットワーク管理ソフトウェアの内部構成を示すブロック図である。

【図5】電源断タスクデータのデータ構成を模式化した図である。

【図6】電源断タスク作成モジュール 421がネットワーク経由でタブレットにより情報処理装置 103上のウェブブラウザに表示し管理者に電源断タスクを作成させることができる電源断タスク作成画面を模式化した図である。

【図7】デバイス設定タスクデータのデータ構成を模式化した図である。

【図8】デバイス設定タスク作成モジュール 431がネットワーク経由でタブレットにより情報処理装置 103上のウェブブラウザに表示し管理者にデバイス設定タスクを作成させることができるデバイス設定タスク作成画面を模式化した図である。

【図9】エラー画面を模式的に示す図である。

【図10】電源断タスク作成モジュール 421が電源断タスクを作成する手順の一例を示すフローチャートである。

【図11】タイマ部 411が処理するスケジュールデータを模式化した図である。

【図12】電源断タスク登録モジュール 424が電源断タスクを登録する手順の一例を示

10

20

30

40

50

すフローチャートである。

【図 1 3】デバイス設定タスク作成モジュール 4 3 1 がデバイス設定タスクを作成する手順の一例を示すフローチャートである。

【図 1 4】デバイス設定タスク登録モジュール 4 3 6 のデバイス設定タスクを登録する手順の一例を示すフローチャートである。

【図 1 5】タイマ通知モジュール 4 4 1 がスケジュールされたタスクを呼び出す手順の一例を示すフローチャートである。

【図 1 6 A】タスク制御部 4 1 3 のメイン処理を示すフローチャートである。

【図 1 6 B】タスク制御部 4 1 3 の電源断タスク処理 [ 1 ] を示すフローチャートである。

【図 1 6 C】タスク制御部 4 1 3 のデバイス設定タスク処理 [ 1 ] を示すフローチャートである。

【図 1 6 D】タスク制御部 4 1 3 の電源断タスク処理 [ 2 ] を示すフローチャートである。

【図 1 6 E】タスク制御部 4 1 3 のデバイス設定タスク処理 [ 2 ] を示すフローチャートである。

【図 1 6 F】タスク制御部 4 1 3 のデバイス設定タスク処理 [ 3 ] を示すフローチャートである。

【図 1 6 G】タスク制御部 4 1 3 のリポート処理 [ 1 ] を示すフローチャートである。

【図 1 7】タスク制御部 4 1 3 がネットワーク経由でサブレットにより情報処理装置 1 0 3 上のウェブブラウザに表示し管理者に電源断タスクをキャンセルさせることができる電源断タスクキャンセル画面を模式化した図である。

【図 1 8】実施例 2 のタスク制御部 4 1 3 の電源断タスク処理 [ 2 ] を示すフローチャートである。

【図 1 9】実施例 3 のタスク制御部 4 1 3 の電源断タスク処理 [ 2 ] を示すフローチャートである。

【図 2 0】実施例 2 のネットワークデバイス管理ソフトウェアの内部構成の一例を示すブロック図である。

【図 2 1 A】実施例 4 のタスク制御部 4 1 3 のデバイス設定タスク処理 [ 2 ] を示すフローチャートである。

【図 2 1 B】実施例 4 のタスク制御部 4 1 3 のリポート処理 [ 1 ] を示すフローチャートである。

【図 2 2】実施例 4 の電源断タスクデータのデータ構成を模式化した図である。

【図 2 3】タスク制御部 4 1 3 がネットワーク経由でサブレットにより情報処理装置 1 0 3 上のウェブブラウザに表示し管理者に電源断タスクをキャンセルさせることができるデバイス設定タスクキャンセル画面を模式化した図である。

【図 2 4】実施例 5 の電源断タスク作成モジュール 4 2 1 が電源断タスクを作成する手順の一例を示すフローチャートである。

【図 2 5】実施例 5 の電源断タスク作成モジュール 4 2 1 がネットワーク経由でサブレットにより情報処理装置 1 0 3 上のウェブブラウザに表示し管理者に電源断タスクを作成させることができる電源断タスク作成画面を模式化した図である。

【図 2 6 A】実施例 5 のタスク制御部 4 1 3 のデバイス設定タスク処理 [ 1 ] を示すフローチャートである。

【図 2 6 B】実施例 5 のタスク制御部 4 1 3 のデバイス設定タスク処理 [ 3 ] を示すフローチャートである。

【図 2 7】実施例 6 のデバイス設定タスク作成モジュール 4 3 1 がネットワーク経由でサブレットにより情報処理装置 1 0 3 上のウェブブラウザに表示し管理者にデバイス設定タスクを作成させることができるデバイス設定タスク作成画面を模式化した図である。

【図 2 8】実施例 6 のデバイス設定タスク作成モジュール 4 3 1 がデバイス設定タスクを作成する手順の一例を示すフローチャートである。

10

20

30

40

50

**【発明を実施するための形態】****【0013】**

以下、本発明を実施するための形態について図面を用いて説明する。

**【実施例1】****【0014】**

図1は、本発明の一実施例を示す画像形成装置を適用可能なシステムの一例を示す図である。

図1において、101はネットワークである。102はネットワーク101に接続されたデバイスであるところの画像形成装置(Multifunction Peripheral)(以下MFPと表記する)である。103はネットワーク101に接続された情報処理装置である。

10

**【0015】**

図2は、MFP102の構成の一部を模式化したブロック図である。

図2において、201はMFP102の制御部である。202はコントローラ本体部分である。コントローラ202は、内部に図示しないCPU、ROM、RAM等を有し、CPUがROMに格納されたプログラムを実行することにより、MFP102全体を統括制御する。

**【0016】**

203はHDD(Hard Disk Drive)やSSD(Solid State Drive)等のストレージである。204は操作パネルである。

205はネットワークインタフェースであり、制御部201はこれを通してネットワーク101に接続している。206はエンジンインタフェースであり、この先にはMFP102の画像形成エンジン(不図示)が接続されている。207は電源コントロールインタフェースであり、この先にはMFP102の電源ユニット(不図示)が接続されている。

20

**【0017】**

図3は、情報処理装置103の構成を模式化したブロック図である。

図3において、301は情報処理装置103の本体である。302は情報処理装置103の中央処理装置(以下CPUと表記する)である。303はCPU302に接続されたバスである。304は読み書き可能メモリ(以下RAMと表記する)である。

**【0018】**

305は読み出し専用メモリ(以下ROMと表記する)である。306はネットワークインタフェースであり、情報処理装置(102)はこれを通してネットワーク(101)に接続している。308はHDD(Hard Disk Drive)やSSD(Solid State Drive)等のストレージである。

30

**【0019】**

307は対人インタフェースデバイス(以下HIDと表記する)インタフェースである。309はキーボードやマウスなどの入力デバイスであり、310はディスプレイやスピーカーなどの出力デバイスである。入力デバイス309、出力デバイス310は、HIDインタフェース307に接続されている。

**【0020】**

図4は、MFP102のコントローラ上202上で動作する本発明の実施例1のネットワークデバイス管理システムのネットワーク管理ソフトウェアの内部構成を示すブロック図である。

40

**【0021】**

図4において、401はネットワーク管理ソフトウェア全体である。このネットワーク管理ソフトウェア401は、コントローラ202内部のCPUが内部のROMに格納されたプログラムを実行することにより実現される機能に相当する。

**【0022】**

ネットワーク管理ソフトウェア401において、411はタイマ部である。412はジョブ検知部である。413はタスク制御部である。414は電源断タスク提供部である。415はデバイス設定タスク提供部である。

50

## 【 0 0 2 3 】

電源断タスク提供部 4 1 4 において、4 2 1 は電源断タスク作成モジュールである。4 2 2 は電源断実行モジュールであり、M F P 1 0 2 の電源断（シャットダウン）を行う。4 2 3 は電源断タスクデータ保存モジュールである。4 2 4 は電源断タスク登録モジュールである。なお、これらの詳細は後述する。

## 【 0 0 2 4 】

デバイス設定タスク提供部 4 1 5 において、4 3 1 はデバイス設定タスク作成モジュールである。4 3 2 はデバイス設定値取得モジュールである。4 3 3 はデバイス設定実行モジュールである。4 3 4 は再起動実行モジュールである。4 3 5 はデバイス設定タスクデータ保存モジュールである。4 3 6 はデバイス設定タスク登録モジュールである。なお、  
10

## 【 0 0 2 5 】

タイマ部 4 1 1 において、4 4 1 はタイマ通知モジュールである。4 4 2 はスケジュールデータ保存モジュールである。なお、これらの詳細は後述する。

図 5 は、電源断タスクデータのデータ構成を模式化した図である。

電源断タスクデータは、電源断タスク作成モジュール 4 2 1 により作成される。また、電源断タスクデータは、電源断タスクデータ保存モジュール 4 2 3 に保存される。

以下、電源断タスクデータの構成について説明する。

5 0 1 は即時フラグで、当該タスクを即時実行するかスケジュール実行するかを決定する。5 0 2 はスケジュール実行する場合に設定される開始時刻である。なお、即時実行する場合でも、電源断タスクデータの作成時にその現在時刻を開始時刻 5 0 2 として格納してもよい。5 0 3 はジョブ待機最大待機時間で、電源断を本当に実行する前にジョブ待機を行う際の待機する時間の上限を決定するためのものである。  
20

## 【 0 0 2 6 】

5 0 4 は強制断フラグで、ジョブ待機最大待機時間 5 0 3 を経過してもジョブが存在する場合に強制的に電源断を実行するかどうかを決定するためのものである。この強制断フラグ 5 0 4 により、管理者は、指定期間（開始時刻からジョブ待機最大待機時間が経過するまで）内に待機ジョブの実行が終了しない場合でも電源断タスクを強制的に実行する強制実行指示の有無を指示することができる。

## 【 0 0 2 7 】

図 6 は、電源断タスク作成モジュール 4 2 1 がネットワーク経由でサーバレットにより情報処理装置 1 0 3 上のウェブブラウザに表示し管理者に電源断タスクを作成させることができる電源断タスク作成画面を模式化した図である。

## 【 0 0 2 8 】

図 6 において、6 0 1 は更新ボタンで、本画面での入力に基づいて電源断タスクの設定を更新するためのものである。6 0 2 は即時実行ボタンで、本画面での入力に基づいて電源断タスクを即時実行するためのものである。6 0 3 はキャンセルボタンで、電源断タスクの設定をしないで本画面を終了するためのものである。

## 【 0 0 2 9 】

6 1 1 は、ジョブ待機最大待機時間 5 0 3 を管理者に入力させるためのものである。6 1 2 は強制電源断チェックボックスで、強制断フラグ 5 0 4 の値を管理者に指定させるためのジョブ待機最大待機時間入力欄である。  
40

## 【 0 0 3 0 】

6 2 1 は、スケジュール実行するかどうかを管理者に指定させるためのスケジュール設定チェックボックスである。

6 2 2 は、スケジュール実行する場合の開始時刻 5 0 2 のうち日付部分を管理者に入力させるための開始時刻日付部分入力欄である。6 2 3 は、スケジュール実行する場合の開始時刻 5 0 2 のうち時間部分を管理者に入力させるための開始時刻時間部分入力欄である。  
50

なお、電源断タスク作成モジュール 4 2 1 が電源断タスクを作成する手順の詳細につい

ては後述する図 10 に示す。

【 0 0 3 1 】

図 7 は、デバイス設定タスクデータのデータ構成を模式化した図である。

デバイス設定タスクデータは、デバイス設定タスク作成モジュール 4 3 1 により作成される。またデバイス設定タスクデータはデバイス設定タスクデータ保存モジュール 4 3 5 に保存される。

【 0 0 3 2 】

図 7 において、7 0 1 は即時フラグで、当該タスクを即時実行するかスケジュール実行するかを決定するためのものである。7 0 2 は開始時刻で、スケジュール実行する場合に設定されるものである。

10

【 0 0 3 3 】

7 0 3 は保存先種別で、デバイス設定値取得モジュール 4 3 2 が取得する対象となるデバイス設定値ファイルの保存先の種別を表すものである。7 0 4 は、デバイス設定値取得モジュールが取得する対象となるデバイス設定値ファイルのファイルパスである。デバイス設定値ファイルは、デバイス設定タスクで設定するデバイスの設定項目と設定値を保持するファイルであり、管理者が予め作成し、M F P 1 0 2 からアクセス可能なネットワーク上の保存先（例えば情報処理装置 1 0 3 内の Windows（登録商標）共有ファイル等）に保存しておく。なお、上記保存先種別 7 0 3 には、Windows（登録商標）共有ファイルや、M F P 1 0 2 内のストレージ 2 0 3 等が設定可能である。

【 0 0 3 4 】

20

7 0 5 は、デバイス設定値取得モジュールがデバイス設定値ファイルにアクセスする際に使用するユーザのユーザ名である。7 0 6 は、デバイス設定値取得モジュールがデバイス設定値ファイルにアクセスする際に使用するユーザのパスワードである。

【 0 0 3 5 】

7 0 7 は設定処理開始前ジョブ待機最大待機時間で、デバイス設定タスクにおけるデバイス設定値のインポート処理を開始する前にジョブ待機を行う際の待機する時間の上限を決定するためのものである。7 0 8 は再起動前ジョブ待機最大待機時間で、デバイス設定タスクにおけるデバイス設定値のインポート処理が完了した後に再起動する前にジョブ待機を行う際の待機する時間の上限を決定するためのものである。

【 0 0 3 6 】

30

図 8 は、デバイス設定タスク作成モジュール 4 3 1 がネットワーク経由でタブレットにより情報処理装置 1 0 3 上のウェブブラウザに表示し管理者にデバイス設定タスクを作成させることができるデバイス設定タスク作成画面を模式化した図である。

【 0 0 3 7 】

図 8 において、8 0 1 は更新ボタンで、本画面での入力に基づいてデバイス設定タスクの設定を更新するためのものである。8 0 2 は、本画面での入力に基づいて即時実行ボタンで、デバイス設定タスクを即時実行するためのものである。8 0 3 はキャンセルボタンで、デバイス設定タスクの設定をしないで本画面を終了するためのものである。

【 0 0 3 8 】

8 1 1 は、保存先種別 7 0 3 を選択するための保存先種別指定ドロップダウンボックスである。選択肢としては Windows（登録商標）共有フォルダーと F T P サーバーが選択できるようになっている。

40

【 0 0 3 9 】

8 1 2 は、ファイルパス 7 0 4 を管理者に入力させるファイルパス入力欄である。8 1 3 は、そのファイルパスにアクセス際に使用するユーザのユーザ名 7 0 5 を管理者に入力させるユーザ名入力欄である。8 1 4 は、そのユーザのパスワード 7 0 6 を管理者に入力させるパスワード入力欄である。

【 0 0 4 0 】

8 2 1 は、作成するデバイス設定タスクにおける設定処理開始前ジョブ待機最大待機時間 7 0 7 を管理者に入力させる設定処理開始前ジョブ待機最大待機時間入力欄である。8

50



22は、作成するデバイス設定タスクにおける再起動前ジョブ待機最大待機時間708を管理者に入力させる再起動前ジョブ待機最大待機時間入力欄である。

【0041】

831は、実行スケジュールを設定するかどうかを管理者に指定させるための実行スケジュール設定有効化チェックボックスである。832は、開始時刻702の日付部分を管理者に入力させる開始時刻日付部分入力欄である。833は、開始時刻702の時間部分を管理者に入力させる開始時刻時間部分入力欄である。

なお、デバイス設定タスク作成モジュール431がデバイス設定タスクを作成する手順の詳細については後述する図13に示す。

【0042】

図9は、エラー画面を模式的に示す図である。

電源断タスク作成画面(図6)あるいはデバイス設定タスク作成画面(図8)において、管理者が何らかの整合性に欠けた入力を行ったとする。そのうえで管理者がそのまま更新ボタン601または801、あるいは、即時実行ボタン602または802を押下した場合には、電源断タスク作成モジュール421またはデバイス設定タスク作成モジュール431は、エラーと判断する。

【0043】

電源断タスク作成モジュール421またはデバイス設定タスク作成モジュール431は、その場合に、このエラー画面を、ネットワーク経由でサブレットにより情報処理装置103のウェブブラウザ上に表示させる。

901は確認ボタンであり、このボタンが押下されると、当該エラー画面が閉じられる。

【0044】

以下、図10のフローチャートを用いて、電源断タスク作成処理について説明する。

図10は、電源断タスク作成モジュール421が電源断タスクを作成する手順の一例を示すフローチャートである。このフローチャートの処理は、コントローラ202内部のCPUが内部のROMに格納されたプログラムを実行することにより実現される。

【0045】

まず、情報処理装置103から電源断タスクの作成要求を受けると、電源断タスク作成モジュール421は、S1001から処理を開始する。

続くS1002において、電源断タスク作成モジュール421は、電源断タスク作成画面(図6)を生成し、情報処理装置103へ送信する。情報処理装置103ではウェブブラウザが動作している。当該ウェブブラウザ上で管理者が必要な情報を入力したのち、更新ボタン601または即時実行ボタン602を押下するか、特に情報を入力しなくてもキャンセルボタン603を押下するなどの操作を行うことで、入力データが、情報処理装置103から電源断タスク作成モジュール421に対して送信される。

【0046】

S1003において、電源断タスク作成モジュール421は、上記情報処理装置から送信された入力データを受信する。

そして続くS1004において、電源断タスク作成モジュール421は、上記入力データが整合しているかどうかをチェックする。

そして、入力データが整合していないと判定した場合(S1004でNo)、電源断タスク作成モジュール421は、S1005に処理を進める。

【0047】

S1005では、電源断タスク作成モジュール421は、エラー画面(図9)を生成し、情報処理装置103へ送信する。情報処理装置103上のウェブブラウザがエラー画面(図9)を描画する。管理者がエラー画面の内容を確認し、確認ボタン901を押下すると、情報処理装置103は、確認ボタン901が押下されたという応答のデータを電源断タスク作成モジュール421に対して送信する。

【0048】

10

20

30

40

50

S 1 0 0 6 において、電源断タスク作成モジュール 4 2 1 は、確認ボタン 9 0 1 が押下されたという応答のデータを受信すると、S 1 0 0 2 に戻って処理を継続するように制御する。

【 0 0 4 9 】

一方、上記 S 1 0 0 4 において、入力データが整合していると判定した場合 ( Y e s ) 、電源断タスク作成モジュール 4 2 1 は、S 1 0 0 7 に処理を進める。

S 1 0 0 7 では、電源断タスク作成モジュール 4 2 1 は、上記入力データの内容を確認し、管理者によって押下されたのが更新ボタン 6 0 1 または即時実行ボタン 6 0 2 であるかどうかをチェックする。

【 0 0 5 0 】

そして、押下されたのが更新ボタン 6 0 1 と即時実行ボタン 6 0 2 のどちらでもないとは判定した場合 ( S 1 0 0 7 で N o ) 、キャンセルボタン 6 0 3 が押下されたということであるので、電源断タスク作成モジュール 4 2 1 は、何もせずそのまま S 1 0 1 1 にて、電源断タスク作成処理を終了する。

【 0 0 5 1 】

一方、押下されたのが更新ボタン 6 0 1 または即時実行ボタン 6 0 2 であると判定した場合 ( S 1 0 0 7 で Y e s ) 、電源断タスク作成モジュール 4 2 1 は、S 1 0 0 8 に処理を進める。

【 0 0 5 2 】

S 1 0 0 8 では、電源断タスク作成モジュール 4 2 1 は、上記入力データの内容から電源断タスクデータ ( 図 5 ) を生成する。

次に S 1 0 0 9 において、電源断タスク作成モジュール 4 2 1 は、上記 S 1 0 0 8 で生成した電源断タスクデータを電源断タスクデータ保存モジュール 4 2 3 へ保存する。

【 0 0 5 3 】

さらに、S 1 0 1 0 において、電源断タスク作成モジュール 4 2 1 は、電源断タスク登録モジュール 4 2 4 を呼び出し、上記 S 1 0 0 9 で保存した電源断タスクデータに対応する電源断タスクを登録する。この際、電源断タスク作成モジュール 4 2 1 は、登録するタスクに対応する電源断タスクデータのポインタ ( 電源断タスクデータ保存モジュール 4 2 3 内の保存場所を示す ) を、電源断タスク登録モジュール 4 2 4 に渡す。この登録処理により、電源断タスクのスケジュールデータ ( 図 1 1 ) がスケジュールデータ保存モジュール 4 4 2 に保存される。なお、電源断タスクの登録処理の詳細は後述する図 1 2 に示す。上記電源断タスクの登録が完了したら、電源断タスク作成モジュール 4 2 1 は、S 1 0 1 1 において、電源断タスク作成処理を完了する。

【 0 0 5 4 】

次に、図 1 1 を用いて、スケジュールデータについて説明する。

図 1 1 は、タイマ部 4 1 1 が処理するスケジュールデータを模式化した図である。

図 1 1 において、1 1 0 1 は、スケジュールの設定時刻である。1 1 0 2 は、スケジュールの種別 ( 電源断タスク、デバイス設定タスク ) を示すデータである。1 1 0 3 は、電源断タスクデータあるいはデバイス設定タスクデータへのポインタである。

【 0 0 5 5 】

以下、図 1 2 のフローチャートを用いて、電源断タスク登録処理について説明する。

図 1 2 は、電源断タスク登録モジュール 4 2 4 が電源断タスクを登録する手順の一例を示すフローチャートである。このフローチャートの処理は、コントローラ 2 0 2 内部の CPU が内部の ROM に格納されたプログラムを実行することにより実現される。

【 0 0 5 6 】

まず、電源断タスク作成モジュール 4 2 1 から呼び出されると、電源断タスク登録モジュール 4 2 4 は、S 1 2 0 1 から処理を開始する。なお、電源断タスク登録モジュール 4 2 4 は、呼び出された際に、電源断タスク作成モジュール 4 2 1 から登録するタスクに対応する電源断タスクデータのポインタ ( 電源断タスクデータ保存モジュール 4 2 3 内の保存場所を示す ) を受け取る。

10

20

30

40

50

## 【 0 0 5 7 】

S 1 2 0 2 では、電源断タスク登録モジュール 4 2 4 は、電源断タスク作成モジュール 4 2 1 から指定されたポイントから読み出された電源断タスクデータに基づいて、登録するタスクが、スケジュール実行が指定されたものか、即時実行が指定されたものかを判定する。

## 【 0 0 5 8 】

スケジュール実行が指定されたものであると判定した場合（S 1 2 0 2 で Y e s ）、電源断タスク登録モジュール 4 2 4 は、S 1 2 0 3 に処理を進める。

S 1 2 0 3 では、電源断タスク登録モジュール 4 2 4 は、上記電源断タスク作成モジュール 4 2 1 から指定されたポイントから読み出された電源断タスクデータに基づいて、スケジュールデータ（図 1 1 ）を作成する。

10

## 【 0 0 5 9 】

さらに、S 1 2 0 4 において、電源断タスク登録モジュール 4 2 4 は、上記 S 1 2 0 3 で作成したスケジュールデータを、タイマ部 4 1 1 のスケジュールデータ保存モジュール 4 4 2 へ保存し、S 1 2 0 6 にて処理を完了する。なお、スケジュールデータ保存モジュール 4 4 2 へスケジュールデータが保存されると、設定時刻 1 1 0 1 がタイマ登録される。そして、設定時刻 1 1 0 1 に到達してシグナルが発生すると、タイマ通知モジュール 4 4 1 は、タスク制御部 4 1 3 へ通知を行う。なお、M F P の再起動時にも、スケジュールデータ保存モジュール 4 4 2 へ保存されたスケジュールデータ内の設定時刻 1 1 0 1 がタイマ登録される。なお、タイマ部 4 1 1 への電源断タスクの登録は 1 タスクしかできないものとする。

20

## 【 0 0 6 0 】

一方、上記 S 1 2 0 2 において、スケジュール実行が指定されたものでない（即時実行が指定されたものである）と判定した場合（N o ）、電源断タスク登録モジュール 4 2 4 は、S 1 2 0 5 に処理を進める。

## 【 0 0 6 1 】

S 1 2 0 5 では、即時実行であるので、電源断タスク登録モジュール 4 2 4 は、タスク制御部 4 1 3 に対して、上記指定された電源断タスクデータのポイントを渡し、S 1 2 0 6 にて処理を完了する。

## 【 0 0 6 2 】

30

以下、図 1 3 のフローチャートを用いてデバイス設定タスク作成処理について説明する。

図 1 3 は、デバイス設定タスク作成モジュール 4 3 1 がデバイス設定タスクを作成する手順の一例を示すフローチャートである。このフローチャートの処理は、コントローラ 2 0 2 内部の C P U が内部の R O M に格納されたプログラムを実行することにより実現される。

## 【 0 0 6 3 】

まず、情報処理装置 1 0 3 からデバイス設定タスクの作成要求を受けると、デバイス設定タスク作成モジュール 4 3 1 は、S 1 3 0 1 から処理を開始する。

続く S 1 3 0 2 において、デバイス設定タスク作成モジュール 4 3 1 は、デバイス設定タスク作成画面（図 8 ）を生成し、情報処理装置 1 0 3 へ送信する。情報処理装置 1 0 3 上ではウェブブラウザが動作しており、管理者が必要な情報を入力したのち、更新ボタン 8 0 1 または即時実行ボタン 8 0 2 を押下するか、特に情報を入力しなくてもキャンセルボタン 8 0 3 を押下するなどの操作を行うことで、入力データが、情報処理装置 1 0 3 からデバイス設定タスク作成モジュール 4 3 1 に対して送信される。

40

## 【 0 0 6 4 】

そして、S 1 3 0 3 において、デバイス設定タスク作成モジュール 4 3 1 は、上記情報処理装置から送信された入力データを受信する。

そして続く S 1 3 0 4 において、デバイス設定タスク作成モジュール 4 3 1 は、上記入力データが整合しているかどうかをチェックする。

50

そして、入力データが整合していないと判定した場合（S 1 3 0 4でN o）、デバイス設定タスク作成モジュール4 3 1は、S 1 3 0 5に処理を進める。

【0 0 6 5】

S 1 3 0 5では、デバイス設定タスク作成モジュール4 3 1は、エラー画面（図9）を生成し、情報処理装置1 0 3へ送信する。情報処理装置1 0 3上のウェブブラウザがエラー画面（図9）を描画する。管理者がエラー画面の内容を確認し、確認ボタン9 0 1を押下すると、情報処理装置1 0 3は、確認ボタン9 0 1が押下されたという応答のデータをデバイス設定タスク作成モジュール4 3 1に対して送信する。

【0 0 6 6】

そして、S 1 3 0 6において、デバイス設定タスク作成モジュール4 3 1は、確認ボタン9 0 1が押下されたという応答のデータを受信すると、S 1 3 0 2に戻って処理を継続するように制御する。

【0 0 6 7】

一方、上記S 1 3 0 4において、入力データが整合していると判定した場合（Y e s）、デバイス設定タスク作成モジュール4 3 1は、S 1 3 0 7に処理を進める。

S 1 3 0 7では、デバイス設定タスク作成モジュール4 3 1は、上記入力データの内容を確認し、管理者によって押下されたのが更新ボタン8 0 1または即時実行ボタン8 0 2であるかどうかをチェックする。

【0 0 6 8】

そして、押下されたのが更新ボタン8 0 1と即時実行ボタン8 0 2のどちらでもないとは判定した場合（S 1 3 0 7でN o）、キャンセルボタン8 0 3が押下されたということであるので、デバイス設定タスク作成モジュール4 3 1は、何もせずそのままS 1 3 1 1にて、デバイス設定タスク作成処理を完了する。

【0 0 6 9】

一方、押下されたのが更新ボタン8 0 1または即時実行ボタン8 0 2であると判定した場合（S 1 3 0 7でY e s）、デバイス設定タスク作成モジュール4 3 1は、S 1 3 0 8に処理を進める。

【0 0 7 0】

S 1 3 0 8では、デバイス設定タスク作成モジュール4 3 1は、上記入力データの内容からデバイス設定タスクデータ（図7）を生成する。

次にS 1 3 0 9において、デバイス設定タスク作成モジュール4 3 1は、そのデバイス設定タスクデータをデバイス設定タスクデータ保存モジュール4 3 5へ保存する。

【0 0 7 1】

さらに、S 1 3 1 0において、デバイス設定タスク作成モジュール4 3 1は、デバイス設定タスク登録モジュール4 3 6を呼び出し、上記S 1 3 0 9でデバイス設定タスクデータ保存モジュール4 3 5に保存したデバイス設定タスクデータに対応するデバイス設定タスクを登録する。この際、デバイス設定タスク作成モジュール4 3 1は、登録するタスクに対応するデバイス設定タスクデータのポインタ（デバイス設定タスクデータ保存モジュール4 3 5内の保存場所を示す）を、デバイス設定タスク登録モジュール4 3 6に渡す。この登録処理により、デバイス設定タスクのスケジュールデータ（図1 1）がスケジュールデータ保存モジュール4 4 2に保存される。なお、デバイス設定タスクの登録処理の詳細は後述する図1 4に示す。上記デバイス設定タスクの登録が完了したら、デバイス設定タスク作成モジュール4 3 1は、S 1 3 1 1において、デバイス設定タスク作成処理を完了する。

【0 0 7 2】

以下、図1 4のフローチャートを参照して、デバイス設定タスク登録処理について説明する。

図1 4は、デバイス設定タスク登録モジュール4 3 6のデバイス設定タスクを登録する手順の一例を示すフローチャートである。このフローチャートの処理は、コントローラ2 0 2内部のCPUが内部のROMに格納されたプログラムを実行することにより実現され

10

20

30

40

50

る。

【 0 0 7 3 】

まず、デバイス設定タスク作成モジュール 4 3 1 から呼び出されると、デバイス設定タスク登録モジュール 4 3 6 は、S 1 4 0 1 から処理を開始する。なお、デバイス設定タスク登録モジュール 4 3 6 は、呼び出された際に、デバイス設定タスク作成モジュール 4 3 1 から登録するタスクに対応するデバイス設定タスクデータのポインタ（デバイス設定タスクデータ保存モジュール 4 3 5 内の保存場所を示す）を受け取る。

【 0 0 7 4 】

S 1 4 0 2 では、デバイス設定タスク登録モジュール 4 3 6 は、デバイス設定タスク作成モジュール 4 3 1 から指定されたポインタから読み出されたデバイス設定タスクデータに基づいて、登録するタスクが、スケジュール実行が指定されたものか、即時実行が指定されたものかを判定する。

10

【 0 0 7 5 】

スケジュール実行が指定されたものであると判定した場合（S 1 4 0 2 で Y e s ）、デバイス設定タスク登録モジュール 4 3 6 は、S 1 4 0 3 に処理を進める。

S 1 4 0 3 では、デバイス設定タスク登録モジュール 4 3 6 は、上記デバイス設定タスク作成モジュール 4 3 1 から指定されたポインタから読み出されたデバイス設定タスクデータに基づいて、スケジュールデータ（図 1 1 ）を作成する。

【 0 0 7 6 】

さらに、S 1 4 0 4 において、デバイス設定タスク登録モジュール 4 3 6 は、上記 S 1 4 0 3 で作成したスケジュールデータを、タイマ部 4 1 1 のスケジュールデータ保存モジュール 4 4 2 へ保存し、S 1 4 0 6 にて処理を完了する。なお、スケジュールデータ保存モジュール 4 4 2 へスケジュールデータが保存されると、設定時刻 1 1 0 1 がタイマ登録される。そして、設定時刻 1 1 0 1 に到達してシグナルが発生すると、タイマ通知モジュール 4 4 1 は、タスク制御部 4 1 3 へ通知を行う。なお、M F P の再起動時にも、スケジュールデータ保存モジュール 4 4 2 へ保存されたスケジュールデータ内の設定時刻 1 1 0 1 がタイマ登録される。なお、タイマ部 4 1 1 へのデバイス設定タスクの登録は 1 タスクしかできないものとする。

20

【 0 0 7 7 】

一方、上記 S 1 4 0 2 において、スケジュール実行が指定されたものでない（即時実行が指定されたものである）と判定した場合（N o ）、デバイス設定タスク登録モジュール 4 3 6 は、S 1 4 0 5 に処理を進める。

30

【 0 0 7 8 】

S 1 4 0 5 では、即時実行であるので、デバイス設定タスク登録モジュール 4 3 6 は、タスク制御部 4 1 3 に対して、上記指定されたデバイス設定タスクデータのポインタを渡し、S 1 4 0 6 にて処理を完了する。

【 0 0 7 9 】

次に、図 1 5 のフローチャートを参照して、スケジュールされたタスクの呼び出し処理について説明する。

図 1 5 は、タイマ通知モジュール 4 4 1 がスケジュールされたタスクを呼び出す手順の一例を示すフローチャートである。このフローチャートの処理は、コントローラ 2 0 2 内部の C P U が内部の R O M に格納されたプログラムを実行することにより実現される。

40

【 0 0 8 0 】

タイマ通知モジュール 4 4 1 は、起動時及び所定のタイミングで呼び出され、S 1 5 0 1 から処理を開始する。

まず、S 1 5 0 2 において、タイマ通知モジュール 4 4 1 は、起動直後であるかどうかをチェックする。

そして、起動直後であると判定した場合（S 1 5 0 2 で Y e s ）、タイマ通知モジュール 4 4 1 は、S 1 5 0 3 へ処理を進める。

S 1 5 0 3 では、タイマ通知モジュール 4 4 1 は、次回起動時指定のスケジュールがあ

50

るか、即ち、設定時刻に「次回起動時」を意味する特別な時刻値（９９：９９）を持つスケジュールデータ（図１１）がスケジュールデータ保存モジュール４４２に保持されているかどうかを判定する。

【００８１】

そして、次回起動時指定のスケジュールがない、即ち、設定時刻に「次回起動時」を意味する特別な時刻値（９９：９９）を持つスケジュールデータがスケジュールデータ保存モジュール４４２に保持されていないと判定した場合（Ｓ１５０３でＮｏ）、タイマ通知モジュール４４１は、Ｓ１５０４に処理を進める。

【００８２】

また、上記Ｓ１５０２において、起動直後でないと判定した場合も（Ｎｏ）、タイマ通知モジュール４４１は、Ｓ１５０４へ処理を進める。

Ｓ１５０４では、タイマ通知モジュール４４１は、スケジュールデータに登録された時刻かどうか、即ち、現在時刻がスケジュールデータ保存モジュール４４２に保存されている各スケジュールデータ（図１１）に保持されている設定時刻１１０１であるかどうかをチェックする。なお、ここでは各スケジュールデータに保持されている設定時刻１１０１を保存時に予めタイマ登録しておきタイマからシグナルを受け取ることで設定時刻１１０１に到達したことが判明するようにシステムが構成されている。

【００８３】

そして、スケジュールデータに登録された時刻でない、即ち、現在時刻がスケジュールデータにある設定時刻１１０１に到達していないと判定した場合（Ｓ１５０４でＮｏ）、タイマ通知モジュール４４１は、再びＳ１５０４へ戻して処理を続行される。

【００８４】

一方、スケジュールデータに登録された時刻である、即ち、現在時刻がスケジュールデータにある設定時刻１１０１に到達したと判定した場合（Ｓ１５０４でＹｅｓ）、タイマ通知モジュール４４１は、Ｓ１５０５に処理を進める。

【００８５】

また、上記Ｓ１５０３で、次回起動時指定のスケジュールがある、即ち、設定時刻に「次回起動時」を意味する特別な時刻値（９９：９９）を持つスケジュールデータがスケジュールデータ保存モジュール４４２に保持されていると判定した場合も（Ｙｅｓ）、タイマ通知モジュール４４１は、Ｓ１５０５に処理を進める。

【００８６】

Ｓ１５０５では、タイマ通知モジュール４４１は、上記Ｓ１５０４で登録時刻となったと判定されたスケジュールデータ、又は、上記Ｓ１５０３で次回起動時指定と判定されたスケジュールを取り出す。

【００８７】

次に、Ｓ１５０６において、タイマ通知モジュール４４１は、上記Ｓ１５０５で取り出したスケジュールデータに保持されているタスクデータへのポインタ１１０３を取り出す。

【００８８】

さらに、Ｓ１５０７において、タイマ通知モジュール４４１は、上記Ｓ１５０６で取り出したタスクデータへのポインタ１１０３を、タスク制御部４１３に対して渡し、Ｓ１５０８にて処理を終了する。これにより、タスクが呼び出されて処理される。

【００８９】

以下、図１６Ａ～図１６Ｇのフローチャートを用いて、タスク制御部４１３による電源断タスクおよびデバイス設定タスクの処理について説明する。

図１６Ａは、タスク制御部４１３のメイン処理を示すフローチャートである。このフローチャートの処理は、コントローラ２０２内部のＣＰＵが内部のＲＯＭに格納されたプログラムを実行することにより実現される。

【００９０】

タイマ通知モジュール４４１、電源断タスク登録モジュール４２４、又はデバイス設定

10

20

30

40

50

タスク登録モジュール436からタスクデータへのポインタ1103を渡されると、タスク制御部413は、S16001から処理を開始する。

【0091】

続くS16002において、タスク制御部413は、上記渡されたポインタに対応するタスクデータを読み出し、該タスクデータが電源断タスクのデータであるかどうかをチェックする。

【0092】

そして、上記渡されたポインタに対応するタスクデータが電源断タスクのデータであると判定した場合（S16002でYes）、タスク制御部413は、S16003において、電源断タスク処理[1]（図16B）を実行する。そして、S16003の電源断タスク処理[1]を抜けたら、タスク制御部413は、S16007にて処理を終了する。

10

【0093】

一方、上記渡されたポインタに対応するタスクデータが電源断タスクのデータでないと判定した場合（S16002でNo）、デバイス設定タスクデータであるので、タスク制御部413は、S16004に処理を進める。

【0094】

S16004では、タスク制御部413は、上記渡されたポインタに対応するデバイス設定タスクデータの開始時刻702を参照し、該開始時刻702から予め定められた時間Ti内の時刻（近い時刻）に、電源断タスクが存在しているかどうかを、電源断タスクデータ保存モジュール423を参照してチェックする。

20

【0095】

そして、上記デバイス設定タスクデータの開始時刻702から時間Ti内の時刻（近い時刻）に電源断タスクが存在していないと判定した場合（S16004でNo）、タスク制御部413は、S16005において、デバイス設定タスク処理[1]（図16C）を実行する。そして、S16005のデバイス設定タスク処理[1]を抜けたら、タスク制御部413は、S16007にて処理を終了する。

【0096】

一方、上記デバイス設定タスクデータの開始時刻702から時間Ti内の時刻（近い時刻）に電源断タスクが存在すると判定した場合（S16004でYes）、タスク制御部413は、S16006において、デバイス設定タスク処理[2]（図16E）を実行する。そして、S16006のデバイス設定タスク処理[2]を抜けたら、タスク制御部413は、S16007にて処理を終了する。

30

【0097】

次に、電源断タスク処理[1]について説明する。

図16Bは、タスク制御部413の電源断タスク処理[1]を示すフローチャートである。このフローチャートの処理は、コントローラ202内部のCPUが内部のROMに格納されたプログラムを実行することにより実現される。

【0098】

電源断タスク処理[1]では、タスク制御部413は、まずS16101において、ジョブ待機（電源断待機）を実行する。このステップは、現在処理中の電源断タスクの電源断のジョブ待機最大待機時間503だけの時間が経過するか、この間にジョブがなくなるか、あるいはこの間に新たなデバイス設定タスクが投入されるかまでの間待機を行う。

40

【0099】

上記S16101のジョブ待機中に、新たなデバイス設定タスクが投入されたと判定した場合、タスク制御部413は、S16102に処理を進める。なお、「新たなデバイス設定タスクが投入された場合」とは、新たなデバイス設定タスクデータのポインタがタスク制御部413に渡された（図15のS1507や図14のS1405）場合である。

【0100】

S16102では、タスク制御部413は、デバイス設定タスクを再スケジュールする。詳細には、タスク制御部413は、上記新たに投入されたデバイス設定タスクのタスク

50

データの内容を使用して、新たに、開始時刻 7 0 2 に次回起動時を示す特別な時刻値「9 9 : 9 9」を指定したデバイス設定タスクデータを作成する。さらにタスク制御部 4 1 3 は、該作成したデバイス設定タスクデータをデバイス設定タスク登録モジュール 4 3 6 により登録する。該登録後、タスク制御部 4 1 3 は、S 1 6 1 0 3 に処理を進める。

【0 1 0 1】

S 1 6 1 0 3 では、タスク制御部 4 1 3 は、電源断タスク処理 [ 2 ] ( 図 1 6 D ) を実行する。そして、S 1 6 1 0 3 の電源断タスク処理 [ 2 ] を抜けたら、タスク制御部 4 1 3 は、電源断タスク処理 [ 1 ] を抜けて、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

【0 1 0 2】

10

また、上記 S 1 6 1 0 1 のジョブ待機中に、ジョブがなくなると判定した場合、タスク制御部 4 1 3 は、S 1 6 1 0 4 に処理を進める。なお、上記 S 1 6 1 0 1 において初めからジョブが無かった場合にも、タスク制御部 4 1 3 は、S 1 6 1 0 4 に処理を進める。

【0 1 0 3】

S 1 6 1 0 4 では、タスク制御部 4 1 3 は、電源断実行モジュール 4 2 2 に電源断を指示する。そして、タスク制御部 4 1 3 は、電源断タスク処理 [ 1 ] を抜けて、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

【0 1 0 4】

また、上記 S 1 6 1 0 1 のジョブ待機中に、ジョブ待機最大待機時間 5 0 3 が経過したと判定した場合、タスク制御部 4 1 3 は、S 1 6 1 0 5 に処理を進める。この場合は、ジョブ待機最大待機時間 5 0 3 が経過しても全てのジョブを終了できなかった場合に対応する。

20

【0 1 0 5】

S 1 6 1 0 5 では、タスク制御部 4 1 3 は、当該電源断タスクデータの強制断フラグ 5 0 4 が立っているかどうかをチェックする。

そして、当該電源断タスクデータの強制断フラグ 5 0 4 が立っていると判定した場合 ( S 1 6 1 0 5 で Y e s )、タスク制御部 4 1 3 は、S 1 6 1 0 4 に処理を進め、電源断実行モジュール 4 2 2 に電源断を指示する。

【0 1 0 6】

一方、当該電源断タスクデータの強制断フラグ 5 0 4 が立っていないと判定した場合 ( S 1 6 1 0 5 で N o )、タスク制御部 4 1 3 は、そのまま電源断タスク処理 [ 1 ] を抜けて、呼び出し元の処理の流れに沿って次のステップへ処理を進める。即ち、この場合、ユーザの意図 ( 強制断フラグ 5 0 4 の設定 ) に従って電源断は実行されないこととなる。

30

【0 1 0 7】

次に、デバイス設定タスク処理 [ 1 ] について説明する。

図 1 6 C は、タスク制御部 4 1 3 のデバイス設定タスク処理 [ 1 ] を示すフローチャートである。このフローチャートの処理は、コントローラ 2 0 2 内部の C P U が内部の R O M に格納されたプログラムを実行することにより実現される。

【0 1 0 8】

デバイス設定タスク処理 [ 1 ] では、タスク制御部 4 1 3 は、まず S 1 6 2 0 1 において、開始前待機を実行する。このステップは、現在処理中のデバイス設定タスクの開始前ジョブ待機最大待機時間 7 0 7 だけの時間が経過するか、この間にジョブがなくなるか、あるいはこの間に新たな電源断タスクが投入されるかまでの間待機を行う。

40

【0 1 0 9】

上記 S 1 6 2 0 1 の開始前待機中に、新たな電源断タスクが投入されたと判定した場合、タスク制御部 4 1 3 は、S 1 6 2 0 2 に処理を進める。なお、「新たな電源断タスクが投入された場合」とは、新たな電源断タスクデータのポインタがタスク制御部 4 1 3 に渡された ( 図 1 5 の S 1 5 0 7 や図 1 2 の S 1 2 0 5 ) 場合である。

【0 1 1 0】

S 1 6 2 0 2 では、タスク制御部 4 1 3 は、現在処理しているデバイス設定タスクのデ

50



バイス設定処理完了想定最終時刻  $T_{comp}$  と電源断タスクの最終実行時刻  $T_s$  とを比較する。なお、上記新たな電源断タスクの開始時刻からジョブ待機最大待機時間 503 だけ経過した時刻を電源断タスクの最終実行時刻  $T_s$  とする。また、上記処理中のデバイス設定タスクの開始時刻から開始前ジョブ待機最大待機時間 707 が経過した時刻をデバイス設定処理開始最終時刻  $T_w$  とする。加えて、デバイス設定値取得モジュール 432 がデバイス設定値ファイルを取得するのに想定される最大必要な時間を  $T_f$  とする。さらに、デバイス設定実行モジュール 433 がデバイス設定を実行開始してから完了するまでに想定される最大必要な時間を  $T_c$  とする。このとき、デバイス設定実行モジュールがデバイス設定の実行を完了するのに想定されるデバイス設定処理完了想定最終時刻  $T_{comp}$  は  $T_w + T_f + T_c$  で算出される。

10

**【0111】**

上記 S16202 において、 $T_{comp}$  が大きい、すなわち電源断タスクの最終実行時刻  $T_s$  の方が先に訪れると判定した場合（「 $T_s$  が先」の場合）、タスク制御部 413 は、S16203 に処理を進める。

**【0112】**

S16203 では、タスク制御部 413 は、デバイス設定タスクを再スケジュールする。詳細には、タスク制御部 413 は、現在処理しているデバイス設定タスクのタスクデータの内容を使用して、新たに、開始時刻 702 に次回起動時を示す特別な時刻値「99:99」を指定したデバイス設定タスクデータを作成する。さらに、タスク制御部 413 は、該作成したデバイス設定タスクデータをデバイス設定タスク登録モジュール 436 により登録する。該登録後、タスク制御部 413 は、S16204 に処理を進める。

20

**【0113】**

S16204 では、タスク制御部 413 は、電源断タスク処理 [2]（図 16D）を実行する。そして、S16204 の電源断タスク処理 [2] を抜けたら、タスク制御部 413 は、デバイス設定タスク処理 [1] を抜けて、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

**【0114】**

また、上記 S16202 において  $T_s$  が大きい、すなわちデバイス設定処理完了想定最終時刻  $T_{comp}$  の方が先に訪れると判定した場合（「 $T_{comp}$  が先」の場合）、タスク制御部 413 は、S16205 に処理を進める。

30

**【0115】**

S16205 では、タスク制御部 413 は、デバイス設定タスク処理 [3]（図 16F）を実行する。なお、S16205 のデバイス設定タスク処理 [3] を抜けたら、デバイス設定タスク処理 [1] を抜けて、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

**【0116】**

また、上記 S16201 の開始前待機中に、ジョブがなくなると判定した場合（「ジョブがなくなった」の場合）、タスク制御部 413 は、S16206 に処理を進める。

S16206 では、タスク制御部 413 は、デバイス設定を実行する。デバイス設定の実行では、タスク制御部 413 は、まずデバイス設定値取得モジュール 432 に、デバイス設定値ファイルを取得させる。次に、タスク制御部 413 は、上記取得したデバイス設定値ファイルを用い、デバイス設定実行モジュール 433 に、デバイス設定を実行させる。

40

**【0117】**

上記 S16206 のデバイス設定実行が完了したら、タスク制御部 413 は、S16207 において、リポート処理 [1]（図 16G）を実行する。そして、S16206 のリポート処理 [1] を抜けたら、タスク制御部 413 は、デバイス設定タスク処理 [1] を抜けて、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

**【0118】**

また、上記 S16201 の開始前待機中に、開始前ジョブ待機最大待機時間 707 を経

50

過したと判定した場合（「待機時間が経過した」の場合）、タスク制御部 4 1 3 は、そのままデバイス設定タスク処理 [ 1 ] を抜けて、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

【 0 1 1 9 】

次に、電源断タスク処理 [ 2 ] について説明する。

図 1 6 D は、タスク制御部 4 1 3 の電源断タスク処理 [ 2 ] を示すフローチャートである。このフローチャートの処理は、コントローラ 2 0 2 内部の C P U が内部の R O M に格納されたプログラムを実行することにより実現される。

【 0 1 2 0 】

電源断タスク処理 [ 2 ] では、タスク制御部 4 1 3 は、まず S 1 6 3 0 1 において、ジョブ待機を実行する。このステップでは、現在処理している電源断タスクのタスクデータのジョブ待機最大待機時間 5 0 3 が経過するか、この間にジョブがなくなるかのいずれかまでの待機を行う。なお、「ジョブ待機最大待機時間 5 0 3 の経過」は、現在処理している電源断タスクの開始時刻からジョブ待機最大待機時間 5 0 3 の経過とする。

10

【 0 1 2 1 】

上記 S 1 6 3 0 1 のジョブ待機中に、ジョブがなくなったと判定した場合、タスク制御部 4 1 3 は、S 1 6 3 0 2 に処理を進める。

S 1 6 3 0 2 では、タスク制御部 4 1 3 は、電源断実行モジュール 4 2 2 へ電源断を指示する。そして、指示が完了したら、タスク制御部 4 1 3 は、電源断タスク処理 [ 2 ] を抜け、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

20

【 0 1 2 2 】

また、上記 S 1 6 3 0 1 のジョブ待機中に、ジョブ待機最大待機時間 5 0 3 が経過したと判定した場合、タスク制御部 4 1 3 は、S 1 6 3 0 3 に処理を進める。

S 1 6 3 0 3 では、タスク制御部 4 1 3 は、強制断フラグ 5 0 4 が立っているかどうかをチェックする。

そして、強制断フラグ 5 0 4 が立っていると判定した場合（S 1 6 3 0 3 で Y e s ）、タスク制御部 4 1 3 は、S 1 6 3 0 2 に処理を進め、電源断実行モジュール 4 2 2 へ電源断を指示する。

【 0 1 2 3 】

一方、強制断フラグ 5 0 4 が立っていないと判定した場合（S 1 6 3 0 3 で N o ）、タスク制御部 4 1 3 は、そのまま電源断タスク処理 [ 2 ] を抜け、呼び出し元の処理の流れに沿って次のステップへ処理を進める。即ち、この場合、ユーザの意図（強制断フラグ 5 0 4 の設定）に従って電源断は実行されないこととなる。

30

【 0 1 2 4 】

次に、デバイス設定タスク処理 [ 2 ] について説明する。

図 1 6 E は、タスク制御部 4 1 3 のデバイス設定タスク処理 [ 2 ] を示すフローチャートである。このフローチャートの処理は、コントローラ 2 0 2 内部の C P U が内部の R O M に格納されたプログラムを実行することにより実現される。

【 0 1 2 5 】

デバイス設定タスク処理 [ 2 ] では、タスク制御部 4 1 3 は、まず S 1 6 4 0 1 において、現在処理しているデバイス設定タスクと、近い時刻に存在する電源断タスクの間で T c o m p と T s とを比較する。

40

【 0 1 2 6 】

そして、T c o m p が小さい、すなわちデバイス設定処理完了想定最終時刻 T c o m pの方が先に訪れると判定した場合（S 1 6 4 0 1 で「T c o m p が先」の場合）、タスク制御部 4 1 3 は、S 1 6 4 0 2 に処理を進める。

【 0 1 2 7 】

S 1 6 4 0 2 では、タスク制御部 4 1 3 は、開始前待機を実行する。このステップでは、処理中のデバイス設定タスクの開始前ジョブ待機最大待機時間 7 0 7 だけの時間が経過するか、この間にジョブがなくなるかまでの間待機を行う。なお、「開始前ジョブ待機最

50

大待機時間 707 の経過」は、処理中のデバイス設定タスクの開始時刻から開始前ジョブ待機最大待機時間 707 の経過とする。

【0128】

そして、上記 S16402 の開始前待機中に、ジョブがなくなつたと判定した場合（「ジョブがなくなつた」の場合）、タスク制御部 413 は、S16403 に処理を進める。

S16403 では、タスク制御部 413 は、デバイス設定を実行する。デバイス設定の実行では、タスク制御部 413 は、まずデバイス設定値取得モジュール 432 に、デバイス設定値ファイルを取得させる。次に、タスク制御部 413 は、上記取得したデバイス設定値ファイルを用い、デバイス設定実行モジュール 433 に、デバイス設定を実行させる。そして、デバイス設定を実行した後、タスク制御部 413 は、S16404 に処理を進める。

10

【0129】

S16404 では、タスク制御部 413 は、上記電源断タスクの最終実行時刻  $T_s$  と現在時刻から予め与えられた再起動省略判定基準時間  $T_n$ （例えば、再起動前ジョブ待機最大待機時間 708 + 再起動実行時間 + デバイスの起動時間に対応する時間）だけ経過した時刻  $T_r$ （以下、再起動完了想定最終時刻  $T_r$  と呼ぶ）との比較を行う。

【0130】

そして、再起動完了想定最終時刻  $T_r$  が小さい（ $T_r$  が先）と判定した場合、すなわち  $T_s$  が再起動省略判定基準を超えている場合、タスク制御部 413 は、再起動を優先して実行するために、S16405 に処理を進める。

20

【0131】

S16405 では、タスク制御部 413 は、電源断タスクを再スケジューリングする。詳細には、タスク制御部 413 は、上記電源断タスクを、該電源断タスクのタスクデータの内容を使用して、新たに、開始時刻 502 に次回起動時実行を意味する「99:99」を指定した新たな電源断タスクデータを作成して登録したのちに消去する。そして、タスク制御部 413 は、S16406 に処理を進める。

【0132】

S16406 では、タスク制御部 413 は、再起動前待機を行う。再起動前待機では、ジョブがなくなるか、処理中のデバイス設定タスクの再起動前ジョブ待機最大待機時間 708 が経過するまでの間待機する。そして、ジョブがなくなるか、または再起動前ジョブ待機最大待機時間 708 が経過したら、タスク制御部 413 は、S16407 に処理を進める。

30

【0133】

S16407 では、タスク制御部 413 は、再起動実行モジュール 434 へ再起動を指示し、デバイス設定タスク処理 [2] を抜け、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

【0134】

また、上記 S16401 において  $T_s$  が先、すなわち電源断タスクの最終実行時刻  $T_s$  の方が先に訪れると判定した場合、タスク制御部 413 は、S16408 に処理を進める。

40

【0135】

S16408 では、タスク制御部 413 は、処理中のデバイス設定タスクの再スケジューリングを行う。詳細には、タスク制御部 413 は、処理中のデバイス設定タスクを、該デバイス設定タスクのデータを流用して開始時刻に次回起動時実行を意味する「99:99」を指定した新たなデバイス設定タスクデータを作成して登録したのちに、消去し、再スケジューリングを行う。そして、タスク制御部 413 は、S16409 に処理を進める。

【0136】

また、上記 S16402 の開始前待機中に、開始前ジョブ待機最大待機時間が経過してしまったと判定した場合（「待機時間が経過」の場合）や、上記 S16404 において、電源断タスクの最終実行時刻  $T_s$  の方が先に訪れると判定した場合（「 $T_s$ 」が先）にも

50

、タスク制御部 4 1 3 は、S 1 6 4 0 9 に処理を進める。

【 0 1 3 7 】

S 1 6 4 0 9 では、タスク制御部 4 1 3 は、電源断タスク処理 [ 2 ] を実行する。電源断タスク処理 [ 2 ] を抜けたら、タスク制御部 4 1 3 は、デバイス設定タスク処理 [ 2 ] を抜け、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

【 0 1 3 8 】

次に、デバイス設定タスク処理 [ 3 ] について説明する。

図 1 6 F は、タスク制御部 4 1 3 のデバイス設定タスク処理 [ 3 ] を示すフローチャートである。このフローチャートの処理は、コントローラ 2 0 2 内部の C P U が内部の R O M に格納されたプログラムを実行することにより実現される。

10

【 0 1 3 9 】

デバイス設定タスク処理 [ 3 ] では、タスク制御部 4 1 3 は、まず S 1 6 5 0 1 において、開始前待機を実行する。このステップでは、処理中のデバイス設定タスクの開始前ジョブ待機最大待機時間 7 0 7 が経過するか、この間にジョブがなくなるかまでの間待機を行う。なお、「開始前ジョブ待機最大待機時間 7 0 7 が経過」とは、処理中のデバイス設定タスクの開始時刻から開始前ジョブ待機最大待機時間 7 0 7 までの経過を示す。

【 0 1 4 0 】

そして、上記 S 1 6 5 0 1 の開始前待機中にジョブがなくなると判定した場合（「ジョブがなくなった」の場合）、タスク制御部 4 1 3 は、S 1 6 4 0 2 に処理を進める。

S 1 6 4 0 2 では、タスク制御部 4 1 3 は、デバイス設定を実行する。デバイス設定の実行では、タスク制御部 4 1 3 は、まずデバイス設定値取得モジュール 4 3 2 に、デバイス設定値ファイルを取得させる。次に、タスク制御部 4 1 3 は、上記取得したデバイス設定値ファイルを用い、デバイス設定実行モジュール 4 3 3 に、デバイス設定を実行させる。そして、タスク制御部 4 1 3 は、S 1 6 5 0 3 に処理を進める。

20

【 0 1 4 1 】

S 1 6 5 0 3 では、タスク制御部 4 1 3 は、電源断タスクの最終実行時刻 T s と再起動完了想定最終時刻 T r との比較を行う。

そして、再起動完了想定最終時刻 T r が先に訪れると判定した場合、先に再起動が完了すると判断して、タスク制御部 4 1 3 は、S 1 6 5 0 4 に処理を進める。

S 1 6 5 0 4 では、タスク制御部 4 1 3 は、電源断タスクを次回起動時実行に再スケジュールし、S 1 6 5 0 5 に処理を進める。

30

S 1 6 5 0 5 では、タスク制御部 4 1 3 は、再起動前待機を行う。再起動前待機は、ジョブがなくなるか再起動前ジョブ待機最大待機時間 7 0 8 が経過するまでの間待機を行う。そして、ジョブがなくなったか再起動前ジョブ待機最大待機時間 7 0 8 が経過したら、タスク制御部 4 1 3 は、S 1 6 5 0 6 に処理を進める。

【 0 1 4 2 】

S 1 6 5 0 6 では、タスク制御部 4 1 3 は、再起動実行モジュール 4 3 4 に再起動を指示する。指示が完了したら、タスク制御部 4 1 3 は、デバイス設定タスク処理 [ 3 ] を抜け、呼び出し元の処理の流れに沿って次のステップに処理を進める。

【 0 1 4 3 】

また、上記 S 1 6 5 0 1 の開始前待機において開始前ジョブ待機最大待機時間 7 0 7 が経過したと判定した場合（「待機時間が経過した」の場合）や、上記 S 1 6 5 0 3 において電源断タスクの最終実行時刻 T s が先に訪れると判定した場合（「T s が先」の場合）、タスク制御部 4 1 3 は、S 1 6 5 0 7 に処理を進める。

40

【 0 1 4 4 】

S 1 6 5 0 7 では、タスク制御部 4 1 3 は、電源断タスク処理 [ 2 ] を実行する。電源断タスク処理 [ 2 ] から抜けたら、タスク制御部 4 1 3 は、デバイス設定タスク処理 [ 3 ] を抜け、呼び出し元の処理の流れに沿って次のステップに処理を進める。

【 0 1 4 5 】

次に、リブート処理 [ 1 ] について説明する。

50

図 16 G は、タスク制御部 413 のリブート処理 [ 1 ] を示すフローチャートである。このフローチャートの処理は、コントローラ 202 内部の CPU が内部の ROM に格納されたプログラムを実行することにより実現される。

【 0146 】

リブート処理 [ 1 ] では、タスク制御部 413 は、まず S16601 において、開始前待機を実行する。このステップは、処理中のデバイス設定タスクの開始前ジョブ待機最大待機時間 707 だけの時間が経過するか、この間にジョブがなくなるか、あるいはこの間に新たな電源断タスクが投入されるかまでの間待機を行う。なお、「開始前ジョブ待機最大待機時間 707 が経過」とは、処理中のデバイス設定タスクの開始時刻から開始前ジョブ待機最大待機時間 707 までの経過を示す。

10

【 0147 】

そして、上記 S16601 の再起動前待機中に、新たな電源断タスクが投入されたと判定した場合、タスク制御部 413 は、S16602 に処理を進める。

S16602 では、タスク制御部 413 は、電源断タスクの最終実行時刻  $T_s$  と再起動完了想定最終時刻  $T_r$  との比較を行う。

そして、 $T_r$  が先に訪れると判定した場合 (S16602 で「 $T_r$  が先」)、再起動が先に完了すると判断して、タスク制御部 413 は、S16603 に処理を進める。

S16603 では、タスク制御部 413 は、電源断タスクを次回起動時実行に再スケジュールし、S16604 に処理を進める。

S16604 では、タスク制御部 413 は、再起動前待機を実行する。S16604 の再起動前待機では、タスク制御部 413 は、ジョブがなくなるか再起動前ジョブ待機最大待機時間 708 が経過するまでの間待機を実行する。ジョブがなくなったか再起動前ジョブ待機最大待機時間 708 が経過した場合、タスク制御部 413 は、S16605 に処理を進める。

20

【 0148 】

S16605 では、タスク制御部 413 は、再起動実行モジュール 434 に再起動を指示する。再起動の指示が完了したら、タスク制御部 413 は、リブート処理 [ 1 ] を抜けて、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

【 0149 】

また、上記 S16601 において、ジョブがなくなったか再起動前ジョブ待機最大待機時間 708 が経過したと判定した場合、タスク制御部 413 は、S16606 に処理を進める。

30

【 0150 】

また、上記 S16602 において、 $T_s$  が先に訪れると判定した場合にも、タスク制御部 413 は、S16606 に処理を進める。

S16606 では、タスク制御部 413 は、電源断タスク処理 [ 2 ] を実行する。そして、電源断タスク処理 [ 2 ] を抜けたら、タスク制御部 413 は、リブート処理 [ 1 ] を抜けて、もとの処理の流れに沿って次のステップに処理を進める。

【 0151 】

以上示したように、実施例 1 によれば、デバイス設定とそれに伴う再起動および電源断が重なるように即時実行またはスケジュール設定指示された場合でも、正しく設定と電源断を行うことが可能になる。特に電源断が確実に実行されているべき時刻において、ユーザの意図に反して電源断が実行されずに終わることがないため、外部からの電源供給を断たれる場合においてもデバイスの故障を未然に防ぐことが可能になる。

40

【 0152 】

即ち、本実施例では、電源断タスク提供部 414 による処理 (第 1 指定処理) により、指定期間内に前記画像形成装置をシャットダウンする電源断タスクを指定し、デバイス設定タスク提供部 415 による処理 (第 2 指定処理) により、指定期間内に前記画像形成装置の設定項目の設定を行う設定工程と該設定工程の後に前記画像形成装置を再起動する再起動工程を行う設定タスクを指定するように構成する。さらに、タスク制御部 413 は、

50

電源断タスクが指示された場合、該電源断タスクの最終実行時刻  $T_s$ （即ち、ユーザによる指定期間の後端であり電源断の許される最終時刻）が、デバイス設定タスクの複数の段階（開始前待機、デバイス設定実行、再起起動待機、再起動実行、デバイスの起動）のどの段階に達するかに応じて、処理を変更するように制御する。

【0153】

例えば、電源断タスクの最終実行時刻  $T_s$ （指定期間の後端）が、デバイス設定タスクの開始時刻からデバイス設定処理完了想定最終時刻  $T_{comp}$  までの間（開始前待機からデバイス設定実行完了までの間、即ち、デバイス設定タスクの指定期間の先端から設定工程が完了すると予想される日時までの期間）に含まれる場合、タスク制御部 413 は、デバイス設定タスクを一旦キャンセルするとともに次回起動時に実行されるように再スケジュールリングして、電源断タスクを実行するように制御する。

10

【0154】

また、電源断タスクの最終実行時刻  $T_s$ （指定期間の後端）が、デバイス設定タスクのデバイス設定処理完了想定最終時刻  $T_{comp}$  の直後から再起動完了想定最終時刻  $T_r$  までの間（デバイス設定実行完了の直後からデバイスの起動完了までの間、即ち、設定工程が完了すると予想される日時から再起動工程により再起動される  $MFP102$  の再起動が完了すると予想される日時までの期間）に含まれる場合、タスク制御部 413 は、デバイス設定タスクのうち、デバイス設定を実行し、再起起動待機以降の処理をキャンセルし、電源断タスクを実行するように制御する。即ち、デバイス設定タスクのうち再起動実行をキャンセルし、電源断タスクを実行するように制御する。

20

【0155】

また、電源断タスクの最終実行時刻  $T_s$ （指定期間の後端）が、デバイス設定タスクの再起動完了想定最終時刻  $T_r$  以降である場合（デバイスの再起動完了後である場合、即ち、設定タスクの再起動工程により再起動される  $MFP102$  の再起動が完了すると予想される日時より以降となる場合）、タスク制御部 413 は、デバイス設定タスクのうちデバイス設定を実行し、電源断タスクを一旦キャンセルするとともに次回起動時に実行されるように再スケジュールリングして、デバイス設定タスクのうち再起起動待機以降の処理を実行する。

【実施例 2】

【0156】

上記実施例 1 によれば、デバイス設定とそれに伴う再起動および電源断が重なるような指示のある場合においても正しく処理を行うことができる。しかし、電源断タスクの実行のために、デバイス設定タスクのうち再起動の実行をキャンセルした後に電源断タスクがキャンセルされた場合、デバイス設定が反映されない状態が続く場合がある。

30

【0157】

例えば、強制フラグがオフで、ジョブ待機最大待機時間 503 が経過してもジョブがなくならなかった場合や、所定のユーザ（管理者）により電源断タスクがキャンセル可能な構成において、電源断タスクがキャンセルされた場合等である。

【0158】

このように、デバイス設定後の再起動の実行がキャンセルされたにも関わらず、電源断が実行されないままとなってしまうケースがある。このような状況では、デバイスは本来反映されるべきデバイス設定の実行結果が、再起動が行われていないため反映されていない状態で放置されてしまうことになる。

40

【0159】

本発明の実施例 2 では、 $MFP102$  は管理者に対してデバイス（ $MFP102$ ）で発生した事象について通知する機能を備え、上記のようなデバイスの再起動が行われず放置された状態が発生した場合、管理者へ通知するように構成する。

【0160】

以下、実施例 2 を、図を示して説明を加える。ただし、実施例 1 との差異についてのみ説明し、説明を省略した部分は実施例 1 と同様とする。

50

まず、実施例 2 のネットワークデバイス管理ソフトウェアについて図 20 を用いて説明する。

図 20 は、実施例 2 のネットワークデバイス管理ソフトウェアの内部構成の一例を示すブロック図である。

なお、図 20 の 401 ~ 442 は、図 4 に記載したものと同様であるので説明は省略する。

#### 【0161】

2001 は、ネットワークデバイス管理システムにおいて管理者に通知すべき事象が発生したときに管理者に通知を行う状態通知部である。状態通知部 2001 は、例えば、電源断タスクの存在によりデバイス設定タスクの再起動処理がキャンセルされた後に、その電源断タスクがキャンセルされた場合等に、その旨を管理者の情報処理装置 103 に送信して、電源断タスクがキャンセルされた旨を管理者に通知する。なお、この通知は、状態通知部 2001 が管理者の通知先（メールアドレス等）にメール等の通知手段を用いて通知するように構成する。なお、管理者の通知先（メールアドレス等）は予め MFP102 のストレージ 203 等の記憶装置に格納されているものとする。

#### 【0162】

次に、図 17 を用いて、管理者により電源断タスクをキャンセルするための構成について説明する。

図 17 は、タスク制御部 413 がネットワーク経由でタブレットにより情報処理装置 103 上のウェブブラウザに表示し管理者に電源断タスクをキャンセルさせることができる電源断タスクキャンセル画面を模式化した図である。

#### 【0163】

図 17 において、1701 は電源断タスクキャンセルの確認ボタンである。1702 は電源断タスクキャンセルの中止ボタンである。

タスク制御部 413 は、情報処理装置 103 から電源断タスクのキャンセル要求を受けると、電源断タスクキャンセル画面（図 17）を生成し、情報処理装置 103 へ送信する。

#### 【0164】

情報処理装置 103 ではウェブブラウザが動作している。当該ウェブブラウザ上で電源断タスクキャンセル画面（図 17）が表示され、管理者が、確認ボタン 1701、又は、中止ボタン 1702 を押下する操作を行うことで、操作情報が、情報処理装置 103 からタスク制御部 413 に対して送信される。

#### 【0165】

タスク制御部 413 は、上記情報処理装置から送信された操作情報を受信し、該操作情報を判断する。そして、上記操作情報が確認ボタン 1701 を押下した操作を示す場合、タスク制御部 413 は、電源断タスクをキャンセルする。一方、上記操作情報が中止ボタン 1702 を押下した操作を示す場合、タスク制御部 413 は、電源断タスクのキャンセルは行わない。

#### 【0166】

次に、図 18 を用いて、実施例 2 の電源断タスク処理 [2] について説明する。なお、上述したように、タスク制御部 413 によるタスク処理の他の部分は実施例 1 と同様であるので説明は省略する。

#### 【0167】

図 18 は、実施例 2 のタスク制御部 413 の電源断タスク処理 [2] を示すフローチャートである。このフローチャートの処理は、コントローラ 202 内部の CPU が内部の ROM に格納されたプログラムを実行することにより実現される。

#### 【0168】

実施例 2 の電源断タスク処理 [2] では、タスク制御部 413 は、まず S18001 において、ジョブ待機を実行する。このステップでは、現在処理している電源断タスクのタスクデータのジョブ待機最大待機時間 503 が経過するか、この間にジョブがなくなるか

10

20

30

40

50

、この間に電源断タスクがキャンセルされるかのいずれかの条件が満たされるまで待機する。なお、「ジョブ待機最大待機時間 5 0 3 の経過」は、現在処理している電源断タスクの開始時刻からジョブ待機最大待機時間 5 0 3 の経過とする。

【 0 1 6 9 】

上記 S 1 8 0 0 1 のジョブ待機中に、ジョブがなくなったと判定した場合、タスク制御部 4 1 3 は、S 1 8 0 0 2 に処理を進める。S 1 8 0 0 2 の処理は、図 1 6 D の S 1 6 3 0 2 の処理と同様であるので説明は省略する。

【 0 1 7 0 】

また、上記 S 1 8 0 0 1 のジョブ待機中に、ジョブ待機最大待機時間 5 0 3 が経過したと判定した場合、タスク制御部 4 1 3 は、S 1 8 0 0 3 に処理を進める。

S 1 8 0 0 3 では、タスク制御部 4 1 3 は、強制断フラグ 5 0 4 が立っているかどうかをチェックする。そして、強制断フラグ 5 0 4 が立っていると判定した場合 ( S 1 8 0 0 3 で Y e s ) 、タスク制御部 4 1 3 は、強制実行指示がなされていると判断し、S 1 8 0 0 2 に処理を進める。

【 0 1 7 1 】

一方、強制断フラグ 5 0 4 が立っていないと判定した場合 ( S 1 8 0 0 3 で N o ) 、タスク制御部 4 1 3 は、強制実行指示がなされていないと判断し、S 1 8 0 0 4 に処理を進める。

【 0 1 7 2 】

また、上記 S 1 8 0 0 1 のジョブ待機中に、電源断タスクがキャンセルされと判定した場合にも、タスク制御部 4 1 3 は、S 1 8 0 0 4 に処理を進める。

S 1 8 0 0 4 では、タスク制御部 4 1 3 は、状態通知部 2 0 0 1 を通じて管理者へ、電源断が行われないままタスクの処理が終わっていることを示す通知を行う。そして、上記 S 1 8 0 4 の通知が完了したら、タスク制御部 4 1 3 は、電源断タスク処理 [ 2 ] を抜け、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

【 0 1 7 3 】

以上示したように、実施例 2 によれば、当初電源断タスクがあったためにデバイス設定にともなう再起動がキャンセルされ、且つその後に電源断タスクがキャンセルされた場合についても、正しく処理することが可能になる。

【 0 1 7 4 】

すなわち、強制断フラグ 5 0 4 の設定により電源断タスクが電源断を強制しなかった場合や、電源断タスクが後に管理者等によりキャンセルされ、M F P 1 0 2 の電源が落ちなかった場合には、管理者に対して通知が行われる。

【 0 1 7 5 】

このような構成により、管理者は、速やかにデバイス ( M F P 1 0 2 ) を手動で再起動することができ、速やかにデバイス設定が正しくデバイスに反映され使用可能な状態とすることができる。

【 0 1 7 6 】

なお、タスク制御部 4 1 3 は、以下に示す場合にも、状態通知部 2 0 0 1 を通じて管理者へ通知を行うように構成してもよい。

例えば、図 1 6 B の S 1 6 1 0 5 で N o と判定された場合、電源断が実行されない旨を管理者に通知するように構成してもよい。

また、図 1 6 C の S 1 6 2 0 1 で「待機時間が経過した」と判定された場合、デバイス設定が実行されない旨を管理者に通知するように構成してもよい。

また、図 1 6 F の S 1 6 5 0 1 の開始前待機において開始前ジョブ待機最大待機時間 7 0 7 が経過したと判定した場合、デバイス設定が実行されない旨を管理者に通知するように構成してもよい。

【 0 1 7 7 】

また、デバイス設定が実行された場合、その旨を管理者に通知するように構成してもよい。



さらに、その他、ジョブの実行等のためデバイス設定が実行されなかった場合のように管理者の設定通りに処理が実行されなかった場合、その旨を管理者に通知するように構成してもよい。

【0178】

また、電源断が実行される直前や再起動が実行される直前又は直後等にも、その旨を管理者に通知するように構成してもよい。

このような構成により、管理者は、速やかにデバイス(MFP102)の状態を把握することができ、管理を容易にしたり、速やかに必要な対処を行うことができる。

【実施例3】

【0179】

上述の実施例2では、当初電源断タスクがあったためにデバイス設定にともなう再起動がキャンセルされ、且つその後に電源断タスクがキャンセルされた場合には、管理者に通知を行って、管理者にデバイスの再起動を促すように構成したが、実施例3では、上記のような場合に自動で再起動を行うように構成する。

【0180】

以下、実施例3を、図を示して説明を加える。ただし、実施例2との差異についてのみ説明し、説明を省略した部分は実施例2と同様とする。

次に、図19を用いて、実施例3の電源断タスク処理[2]について説明する。なお、上述したように、タスク制御部413によるタスク処理の他の部分は実施例1と同様であるので説明は省略する。

【0181】

図19は、実施例3のタスク制御部413の電源断タスク処理[2]を示すフローチャートである。このフローチャートの処理は、コントローラ202内部のCPUが内部のROMに格納されたプログラムを実行することにより実現される。なお、図18と同一のステップには同一のステップ番号を付してある。以下、図18と異なる部分のみ説明する。

【0182】

S18003において、強制断フラグ504が立っていないと判定した場合(No)、タスク制御部413は、強制実行指示がなされていないと判断し、待機ジョブの処理が終了した後、S19004に処理を進める。

また、上記S18001のジョブ待機中に、電源断タスクがキャンセルされと判定した場合にも、タスク制御部413は、S19004に処理を進める。

S19004では、タスク制御部413は、再起動実行モジュール434に再起動を指示する。そして、指示が完了したら、タスク制御部413は、電源断タスク処理[2]を抜けて、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

【0183】

以上示したように、実施例3によれば、当初電源断タスクがあったためにデバイス設定にともなう再起動がキャンセルされ、且つその後に電源断タスクがキャンセルされた場合についても、正しく処理することが可能になる。

【0184】

すなわち、強制断フラグ504の設定により電源断タスクが電源断を強制しなかった場合や、電源断タスクが後に管理者等によりキャンセルされ、MFP102の電源が落ちなかった場合には、あらためて再起動が正しく行われ、デバイス設定が正しくデバイスに反映され使用可能な状態になる。

【0185】

上述の実施例2では、管理者が通知を受け取って手動でデバイスを再起動する必要があったが、実施例3においてはその必要がないというメリットも存在する。

なお、実施例3でも、図20で示した状態通知部2001を設け、以下のような場合に、実施例2と同様に、タスク制御部413が、状態通知部2001を通じて管理者へ通知を行うように構成してもよい。

【0186】

10

20

30

40

50

例えば、図 19 の S 1 9 0 0 4 で再起動を行った場合、再起動を実行した旨を管理者に通知するように構成してもよい。

その他、実施例 2 の最後に示した各種の場合にも、管理者に通知するように構成してもよい。

このような構成により、管理者は、速やかにデバイス (M F P 1 0 2) の状態を把握することができ、管理を容易にしたり、速やかに必要な対処を行うことができる。

#### 【実施例 4】

##### 【0187】

上述の実施例 1 ~ 3 においては、電源断タスクの最終実行時刻  $T_s$  がデバイス設定処理完了想定最終時刻  $T_{comp}$  よりも前である場合には必ずデバイス設定タスクを次回起動時に実行するように構成した。この場合、デバイス設定タスクが完了して設定が反映されるのは想定よりも遅くなってしまう。しかし、デバイス設定タスクの実行を優先してしまうと、デバイス設定実行モジュール 433 がデバイス設定を実行している間に電源断タスクの最終実行時刻を過ぎてしまい、電源断を実行できない可能性がある。

##### 【0188】

そこで、実施例 4 では、上記の事情を鑑み、電源断タスクがデバイス設定実行モジュール 433 の実行中に最終実行時刻を迎えるように設定されてしまった場合でも、できるだけデバイス設定タスクと電源断タスクの両方を実行することができるよう構成する。

##### 【0189】

以下、実施例 4 を、図を示して説明を加える。ただし、実施例 1 ~ 3 との差異についてのみ説明し、説明を省略した部分は実施例 1 ~ 3 のいずれかと同様とする。

まず、図 22 を用いて、実施例 4 の電源断タスクデータについて説明する。

図 22 は、実施例 4 の電源断タスクデータのデータ構成を模式化した図である。

図 22 において、501 ~ 504 は図 5 の 501 ~ 504 と同様であるので説明は省略する。2201 はジョブ待機最終時刻である。

このジョブ待機最終時刻 2201 は、図 10 の S 1 0 0 8 において、電源断タスク作成モジュール 421 が、電源断タスクデータを生成する際に以下のような方法で算出する。即時フラグ 501 が「スケジュール実行」を示す場合、開始時刻 502 にジョブ待機最大待機時間 503 を加えることによりジョブ待機最終時刻 2201 を算出される。また、即時フラグ 501 が「即時実行」を示す場合、現在時刻にジョブ待機最大待機時間 503 を加えることによりジョブ待機最終時刻 2201 を算出される。

##### 【0190】

次に、図 21A を用いて、実施例 4 のデバイス設定タスク処理 [2] について説明する。また、図 21B を用いて、実施例 4 のリポート処理 [1] について説明する。なお、上述したように、タスク制御部 413 によるタスク処理の他の部分は実施例 1 ~ 3 のいずれかと同様であるので説明は省略する。

##### 【0191】

図 21A は、実施例 4 のタスク制御部 413 のデバイス設定タスク処理 [2] を示すフローチャートである。このフローチャートの処理は、コントローラ 202 内部の CPU が内部の ROM に格納されたプログラムを実行することにより実現される。なお、図 16E と同一のステップには同一のステップ番号を付してある。以下、図 16E と異なる部分のみ説明する。

##### 【0192】

実施例 4 のデバイス設定タスク処理 [2] では、タスク制御部 413 は、まず S 1 6 4 0 1 において、現在処理しているデバイス設定タスクと、近い時刻に存在する電源断タスクの間でデバイス設定処理完了想定最終時刻  $T_{comp}$  と電源断タスク最終実行時刻  $T_s$  との比較の結果、デバイス設定処理完了想定最終時刻  $T_{comp}$  より電源断タスク最終実行時刻  $T_s$  の方が先に訪れると判定した場合 (S 1 6 4 0 1 で「 $T_s$  が先」の場合)、タスク制御部 413 は、S 2 1 4 0 1 に処理を進める。

##### 【0193】

10

20

30

40

50

S 2 1 4 0 1では、タスク制御部 4 1 3は、電源断タスクの開始時刻 5 0 2と上記 T c o m pを比較する。

そして、開始時刻 5 0 2が T c o m pよりも後であると判定した場合 ( S 2 1 4 0 1で「 T c o m pが先」の場合)、タスク制御部 4 1 3は、何もせず S 1 6 4 0 2に処理を進める。

【 0 1 9 4 】

一方、開始時刻 5 0 2が T c o m pよりも先であると判定した場合 ( S 2 1 4 0 1で「開始時刻 5 0 2が先」の場合)、タスク制御部 4 1 3は、S 2 1 4 0 2に処理を進める。

S 2 1 4 0 2では、タスク制御部 4 1 3は、電源断タスクを再スケジュールする。詳細には、タスク制御部 4 1 3は、電源断タスクをデバイス設定処理完了直後に実行することを意味する開始時刻「 8 8 : 8 8」にセットした電源断タスクデータを作成登録し、もとの電源断タスクデータを消去することで再スケジュールする。再スケジュールが完了したら、タスク制御部 4 1 3は、S 1 6 4 0 2に処理を進める。

【 0 1 9 5 】

また、S 1 6 4 0 3にてデバイス設定が完了したら、タスク制御部 4 1 3は、S 2 1 4 0 3に処理を進める。

S 2 1 4 0 3では、タスク制御部 4 1 3は、開始時刻「 8 8 : 8 8」に指定された電源断タスクが存在するかどうかをチェックすることで、デバイス設定処理完了直後に実行する電源断タスクがあるかどうかを判定する。

【 0 1 9 6 】

そして、開始時刻「 8 8 : 8 8」に指定された電源断タスクが存在しない場合はデバイス設定処理完了直後に実行する電源断タスクがないと判定し ( S 2 1 4 0 3で「なし」)、タスク制御部 4 1 3は、S 1 6 4 0 4に処理を進める。

【 0 1 9 7 】

一方、開始時刻「 8 8 : 8 8」に指定された電源断タスクが存在する場合はデバイス設定処理完了直後に実行する電源断タスクがあると判定し ( S 2 1 4 0 3で「あり」)、タスク制御部 4 1 3は、S 2 1 4 0 4に処理を進める。

【 0 1 9 8 】

S 2 1 4 0 4では、タスク制御部 4 1 3は、待機時間を再計算する。このとき、タスク制御部 4 1 3は、現在時刻が当初の電源断タスクの最終実行時刻 T s (即ち、ジョブ待機最終時刻 2 2 0 1)を過ぎている場合には、開始時刻 5 0 2に現在時刻、ジョブ待機最大待機時間 5 0 3に「 0」をそれぞれセットし、待機時間をゼロとする。一方、現在時刻が当初の電源断タスクの最終実行時刻 T s (ジョブ待機最終時刻 2 2 0 1)を過ぎていない場合には、開始時刻 5 0 2に現在時刻、ジョブ待機最大待機時間 5 0 3に T sと現在時刻の差をそれぞれセットし、待機時間を T sと現在時刻の差になるようにする。以上のような待機時間の再計算が完了したら、タスク制御部 4 1 3は、S 1 6 4 0 9に処理を進める。なお、他のステップは、図 1 6 Eと同一であるので説明は省略する。

【 0 1 9 9 】

図 2 1 Bは、実施例 4のタスク制御部 4 1 3のリポート処理 [ 1 ]を示すフローチャートである。このフローチャートの処理は、コントローラ 2 0 2内部の C P Uが内部の R O Mに格納されたプログラムを実行することにより実現される。なお、図 1 6 Gと同一のステップには同一のステップ番号を付してある。以下、図 1 6 Gと異なる部分のみ説明する。

【 0 2 0 0 】

実施例 4のリポート処理 [ 1 ]では、タスク制御部 4 1 3は、まず S 2 1 6 0 1において、新規電源断タスクがあるかどうかをチェックする。

そして、新規電源断タスクがあると判定した場合 ( S 2 1 6 0 1で「ある」の場合)、タスク制御部 4 1 3は、S 2 1 6 0 2に処理を進める。

S 2 1 6 0 2では、タスク制御部 4 1 3は、現在時刻と電源断タスクの最終実行時刻 T sを比較する。

10

20

30

40

50

## 【 0 2 0 1 】

そして、まだ現在時刻が  $T_s$  を過ぎていないと判定した場合、タスク制御部 4 1 3 は、S 1 6 6 0 2 に処理を進める。

一方、すでに現在時刻が  $T_s$  を過ぎていていると判定した場合、タスク制御部 4 1 3 は、S 1 6 6 0 6 に処理を進める。

## 【 0 2 0 2 】

以上示したように、デバイス設定タスクのデバイス設定処理開始後からデバイス設定処理完了想定最終時刻  $T_{comp}$  までの間に、電源断タスクの最終実行時刻  $T_s$  を迎えてしまうケースが存在する。上述した実施例 1 ~ 3 によれば、そのようなケースではデバイス設定タスクを全てキャンセルしてしまい、次回起動時に実行するため、デバイス設定タスクが最終的に実行される時刻が大幅に後ろにずれてしまうことになる。しかし、実施例 4 によれば、そのような場合でも可能な限りデバイス設定タスクによるデバイス設定を当初意図した時刻に実行することができ、電源断も行うことが可能になる。

## 【 0 2 0 3 】

また、リブート処理 [ 1 ] は「再起動前待機」と「再起動を指示」の 2 ステップに分かれているが、実施例 1 ~ 3 では、まず「再起動前待機」に入ってから新規電源断タスクの存在をチェックしていた。しかし、実施例 4 では、リブート処理 [ 1 ] の直前の「デバイス設定処理」の最中に新たに投入されて最終実行時刻  $T_s$  に到達してしまう電源断タスクが存在する場合には、その電源断タスクを優先的に処理するように構成する。

## 【 0 2 0 4 】

以上示したように、実施例 1 ~ 3 では、デバイス設定タスクのデバイス設定処理完了想定最終時刻  $T_{comp}$  が電源断タスクの最終実行時刻  $T_s$  を過ぎてしまうような場合には必ず「デバイス設定処理」を再起動後に実行していたため、デバイス設定が管理者の想定より遅くなってしまうケースが多くなる可能性があった。しかし、実施例 4 では、電源断タスクの指定期間の後端 ( $T_s$ ) が、デバイス設定タスクの指定期間の先端から設定工程が完了すると予想される日時 ( $T_{comp}$ ) までの期間に含まれる場合であっても、電源断タスクの指定期間の先端 (開始時刻 5 0 2) が、デバイス設定タスクの設定工程が完了すると予想される日時 ( $T_{comp}$ ) よりも先である場合には、電源断タスクをデバイス設定タスクの設定工程の直後に実行するように再スケジュールリングするとともにして、デバイス設定タスクの設定工程を実行するように制御するように構成し、上記のような場合にもデバイス設定処理を実行可能な場合を設けたので、実施例 1 ~ 3 に比べて、デバイス設定処理が実行されるケースを増やすことができる。よって、実施例 1 ~ 3 に比べて、デバイス設定が管理者が想定した時刻に実行できるケースを増やすことが可能となる。

## 【 実施例 5 】

## 【 0 2 0 5 】

実際のシステムでは、デバイス設定タスクによるデバイス設定が実行されている間に外部からの電源供給を断たれてしまうケースが起こり得る。しかし、このようなケースで、管理者が電源供給を断たれてしまう時刻を意識して電源断タスクを作成する時に、デバイス設定タスクが同時刻にデバイス設定処理開始してしまうことに気がつかないことがあり得る。

## 【 0 2 0 6 】

実施例 5 では、そのようなケースでは電源断タスクの作成に制限を設けることで上記のような事態を回避する構成である。

以下、実施例 5 を、図を示して説明を加える。ただし、実施例 1 ~ 4 との差異についてのみ説明し、説明を省略した部分は実施例 1 ~ 4 のいずれかと同様とする。

## 【 0 2 0 7 】

まず、図 2 4 のフローチャートを用いて、実施例 5 の電源断タスク作成処理について説明する。

図 2 4 は、実施例 5 の電源断タスク作成モジュール 4 2 1 が電源断タスクを作成する手順の一例を示すフローチャートである。このフローチャートの処理は、コントローラ 2 0

10

20

30

40

50

2 内部の CPU が内部の ROM に格納されたプログラムを実行することにより実現される。なお、図 10 と同一のステップには同一のステップ番号を付してある。また、ここでは図 10 との差異についてのみ説明する。

【0208】

まず、情報処理装置 103 から電源断タスクの作成要求を受けると、電源断タスク作成モジュール 421 は、S2401 から処理を開始する。

続く S2402 において、電源断タスク作成モジュール 421 は、デバイス設定タスクデータが存在しているかどうかを判定する。

そして、デバイス設定タスクデータが存在していないと判定した場合 (S2402 で No)、電源断タスク作成モジュール 421 は、S1002 に処理を進める。

10

【0209】

そして、デバイス設定タスクデータが存在していると判定した場合 (S2402 で Yes)、電源断タスク作成モジュール 421 は、S2403 に処理を進める。

S2403 では、電源断タスク作成モジュール 421 は、電源断タスクの制約条件を計算する。すなわち、電源断タスク作成モジュール 421 は、電源断タスクの最終実行時刻 Ts がデバイス設定処理完了想定最終時刻 Tcomp より先に到達するような電源断タスクは作成できないように制約条件を計算する。そして、電源断タスク作成モジュール 421 は、S1002 に処理を進める。

【0210】

さらに、S2404 のデータ整合チェック処理において、電源断タスク作成モジュール 421 は、図 10 の S1004 におけるデータの整合性の検証のほかに、上記制約条件に抵触するような電源断タスクを作成しようとするデータも不整合とみなすものとする。なお、他の処理は、図 10 に示した処理と同様であるので説明は省略する。

20

【0211】

以下、図 25 を用いて、実施例 5 の電源断タスク作成画面について説明する。

図 25 は、実施例 5 の電源断タスク作成モジュール 421 がネットワーク経由でサブレットにより情報処理装置 103 上のウェブブラウザに表示し管理者に電源断タスクを作成させることができる電源断タスク作成画面を模式化した図である。なお、図 6 と同一のものには同一の符号を付してある。また、ここでは図 6 との差異についてのみ説明する。

【0212】

図 25 において、2501 は上述の制約条件について管理者へ提示するメッセージである。このメッセージはデバイス設定タスクが存在しない場合には表示されることはない。このためデバイス設定タスクが存在しない場合の電源断タスク作成画面は図 6 に掲げたものと同様になる。

30

【0213】

次に、図 23 を用いて、管理者によりデバイス設定タスクをキャンセルするための構成について説明する。

図 23 は、タスク制御部 413 がネットワーク経由でサブレットにより情報処理装置 103 上のウェブブラウザに表示し管理者に電源断タスクをキャンセルさせることができるデバイス設定タスクキャンセル画面を模式化した図である。

40

【0214】

図 23 において、2301 はデバイス設定タスクキャンセルの確認ボタンである。1702 はデバイス設定タスクキャンセルの中止ボタンである。

タスク制御部 413 は、情報処理装置 103 からデバイス設定タスクのキャンセル要求を受けると、デバイス設定タスクキャンセル画面 (図 23) を生成し、情報処理装置 103 へ送信する。

【0215】

情報処理装置 103 ではウェブブラウザが動作している。当該ウェブブラウザ上でデバイス設定タスクキャンセル画面 (図 23) が表示され、管理者が、確認ボタン 2301、又は、中止ボタン 2302 を押下する操作を行うことで、操作情報が、情報処理装置 10

50

3 からタスク制御部 4 1 3 に対して送信される。

【 0 2 1 6 】

タスク制御部 4 1 3 は、上記情報処理装置から送信された操作情報を受信し、該操作情報を判断する。そして、上記操作情報が確認ボタン 2 3 0 1 を押下した操作を示す場合、タスク制御部 4 1 3 は、デバイス設定タスクをキャンセルする。一方、上記操作情報が中止ボタン 2 3 0 2 を押下した操作を示す場合、タスク制御部 4 1 3 は、デバイス設定タスクのキャンセルは行わない。

【 0 2 1 7 】

次に、図 2 6 A を用いて、実施例 5 のデバイス設定タスク処理 [ 1 ] について説明する。また、図 2 6 B を用いて、実施例 5 のデバイス設定タスク処理 [ 3 ] について説明する。なお、タスク制御部 4 1 3 によるタスク処理の他の部分は実施例 1 ~ 4 のいずれかと同様であるので説明は省略する。

10

【 0 2 1 8 】

図 2 6 A は、実施例 5 のタスク制御部 4 1 3 のデバイス設定タスク処理 [ 1 ] を示すフローチャートである。このフローチャートの処理は、コントローラ 2 0 2 内部の CPU が内部の ROM に格納されたプログラムを実行することにより実現される。なお、図 1 6 C と同一のステップには同一のステップ番号を付してある。以下、図 1 6 C と異なる部分のみ説明する。

【 0 2 1 9 】

実施例 5 のデバイス設定タスク処理 [ 1 ] では、タスク制御部 4 1 3 は、まず S 2 6 2 0 1 において、開始前待機を実行する。このステップは、現在処理中のデバイス設定タスクの開始前ジョブ待機最大待機時間 7 0 7 だけの時間が経過するか、この間にジョブがなくなるか、この間に新たな電源断タスクが投入されるか、あるいはこの間にデバイス設定タスクがキャンセルされるかまでの間待機を行う。

20

【 0 2 2 0 】

上記 S 2 6 2 0 1 の開始前待機中に、新たな電源断タスクが投入されたと判定した場合、タスク制御部 4 1 3 は、S 1 6 2 0 2 に処理を進める。なお、S 1 6 2 0 2 ~ S 1 6 2 0 5 の処理は図 1 6 C と同様であるので説明は省略する。

【 0 2 2 1 】

また、上記 S 2 6 2 0 1 の開始前待機中に、ジョブがなくなったと判定した場合（「ジョブがなくなった」の場合）、タスク制御部 4 1 3 は、S 1 6 2 0 6 に処理を進める。なお、S 1 6 2 0 6 ~ S 1 6 2 0 7 の処理は図 1 6 C と同様であるので説明は省略する。

30

【 0 2 2 2 】

また、上記 S 2 6 2 0 1 の開始前待機中に、開始前ジョブ待機最大待機時間 7 0 7 を経過したと判定した場合（「待機時間が経過した」の場合）、タスク制御部 4 1 3 は、そのままデバイス設定タスク処理 [ 1 ] を抜けて、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

【 0 2 2 3 】

また、上記 S 2 6 2 0 1 の開始前待機中に、デバイス設定タスクがキャンセルされたと判定した場合（「デバイス設定タスクがキャンセルされた」の場合）、タスク制御部 4 1 3 は、そのままデバイス設定タスク処理 [ 1 ] を抜けて、呼び出し元の処理の流れに沿って次のステップへ処理を進める。

40

【 0 2 2 4 】

図 2 6 B は、実施例 5 のタスク制御部 4 1 3 のデバイス設定タスク処理 [ 3 ] を示すフローチャートである。このフローチャートの処理は、コントローラ 2 0 2 内部の CPU が内部の ROM に格納されたプログラムを実行することにより実現される。なお、図 1 6 F と同一のステップには同一のステップ番号を付してある。以下、図 1 6 F と異なる部分のみ説明する。

【 0 2 2 5 】

実施例 5 のデバイス設定タスク処理 [ 3 ] では、タスク制御部 4 1 3 は、まず S 2 6 5

50

01において、開始前待機を実行する。このステップでは、処理中のデバイス設定タスクの開始前ジョブ待機最大待機時間707が経過するか、あるいはこの間にジョブがなくなるか、この間にデバイス設定タスクがキャンセルされるかまでの間待機を行う。

【0226】

そして、上記S26501の開始前待機中にジョブがなくなると判定した場合（「ジョブがなくなった」の場合）、タスク制御部413は、S16402に処理を進める。なお、S16502～S16506の処理は図16Fと同様であるので説明は省略する。

【0227】

また、上記S26501の開始前待機において開始前ジョブ待機最大待機時間707が経過したと判定した場合（「待機時間が経過した」の場合）、タスク制御部413は、S16507に処理を進める。

【0228】

また、上記S26501の開始前待機においてデバイス設定タスクがキャンセルされたと判定した場合（「デバイス設定タスクがキャンセルされた」の場合）、タスク制御部413は、S16507に処理を進める。

なお、S16507の処理は図16Fと同様であるので説明は省略する。

【0229】

以上示したように、実施例5によれば、管理者は、すでにデバイス設定タスクが存在している場合にデバイス設定値取得モジュール432やデバイス設定実行モジュール433がデバイス設定値の取得と設定を実行している間に電源断を行う可能性がある電源断タスクを作成できない。即ち、電源断タスク提供部414では、指定する電源断タスクの指定期間の後端が、指定済みの前記設定タスクの指定期間の先端から設定工程が完了すると予想される日時までの期間に含まれるような、電源断タスクの指定を禁止する。これにより、デバイス設定値の取得と実行を行っている間に電源断がスケジュールされる危険性を管理者に知らせることが可能になる。

【実施例6】

【0230】

上記実施例5においては、デバイス設定タスクは自由に設定できるため、すでに電源断タスクが設定されているときに、該電源断タスクと競合するようなデバイス設定タスクを設定することが可能である。

【0231】

実施例6では、デバイス設定タスクの作成時に、電源断タスクが存在している場合には、該電源断タスクと競合するようなデバイス設定タスクを作成できないような制約条件を設ける構成とする。

以下、実施例6を、図を示して説明を加える。ただし、実施例5との差異についてのみ説明し、説明を省略した部分は実施例5と同様とする。

【0232】

まず、図28のフローチャートを用いて実施例6のデバイス設定タスク作成処理について説明する。

図28は、実施例6のデバイス設定タスク作成モジュール431がデバイス設定タスクを作成する手順の一例を示すフローチャートである。このフローチャートの処理は、コントローラ202内部のCPUが内部のROMに格納されたプログラムを実行することにより実現される。なお、図13と同一のステップには同一のステップ番号を付してある。以下、図13と異なる部分のみ説明する。

【0233】

まず、情報処理装置103からデバイス設定タスクの作成要求を受けると、実施例6のデバイス設定タスク作成モジュール431は、S2801から処理を開始する。

続くS2802において、デバイス設定タスク作成モジュール431は、電源断タスクデータが存在するかどうかをチェックする。

そして、電源断タスクデータが存在すると判定した場合（S2802でYes）、デバ

10

20

30

40

50

イス設定タスク作成モジュール431は、S2803に処理を進める。

【0234】

S2803では、デバイス設定タスク作成モジュール431は、デバイス設定タスクの制約条件を計算する。電源断が起こり得る時間帯、すなわち電源断タスクの開始時刻502から最終実行時刻Tsまでの間にデバイス設定値取得モジュール432またはデバイス設定実行モジュール433が呼ばれるような実行時刻および待機時間の設定はできないように制約条件を計算する。そして、デバイス設定タスク作成モジュール431は、S1302に処理を進める。

【0235】

また、上記S2802において、電源断タスクデータが存在しないと判定した場合(N o)、デバイス設定タスク作成モジュール431は、S1302に処理を進める。

【0236】

さらに、S2804のデータ整合チェック処理において、デバイス設定タスク作成モジュール431は、図13のS1307におけるデータの整合性の検証のほかに、上記制約条件に抵触するようなデバイス設定タスクを作成しようとするデータも不整合とみなすものとする。なお、他の処理は、図13に示した処理と同様であるので説明は省略する。

【0237】

次に、図27を用いて、実施例6のデバイス設定タスク作成画面について説明する。

図27は、実施例6のデバイス設定タスク作成モジュール431がネットワーク経由でタブレットにより情報処理装置103上のウェブブラウザに表示し管理者にデバイス設定タスクを作成させることができるデバイス設定タスク作成画面を模式化した図である。なお、図8と同一のものには同一の符号を付してある。また、ここでは図8と異なる部分についてのみ説明する。

【0238】

図27において、2701は上述の制約条件について管理者へ提示するメッセージである。このメッセージはデバイス設定タスクが存在しない場合には表示されることはない。このためデバイス設定タスクが存在しない場合の電源断タスク作成画面は図8に掲げたものと同様になる。

【0239】

以上説明したように、実施例6によれば、管理者は、すでに電源断タスクが存在している場合に、電源断が行われる可能性がある時刻にデバイス設定値の取得と設定を実行するようなデバイス設定電源断タスクを作成できない。即ち、デバイス設定タスク提供部415では、指定済みの電源断タスクの指定期間の後端が、指定するデバイス設定タスクの指定期間の先端から設定工程が完了すると予想される日時までの期間に含まれないようなデバイス設定タスクの指定を禁止する。これにより電源断を行っている間にデバイス設定値の取得と実行を行うようなスケジュールが設定される危険性を管理者に知らせることが可能になる。

【0240】

以上示したように、本発明の各実施例によれば、デバイス設定とそれに伴う再起動、および、電源断を、重なるようにそれぞれを即時実行またはスケジュール設定して指示された場合においても、正しくデバイス設定と電源断を行うことが可能になる。

【0241】

特に、電源断が確実に実行されているべき時刻において電源断が実行されずに終わることがないため、外部から電源供給を断たれる場合においてもデバイス(MFP102)の故障を未然に防ぐことが可能になる。

【0242】

なお、上述した各種データの構成及びその内容はこれに限定されるものではなく、用途や目的に応じて、様々な構成や内容で構成されることは言うまでもない。

以上、一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

10

20

30

40

50



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。

また、上記各実施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。

#### 【0243】

(他の実施例)

また、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。

#### 【0244】

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。

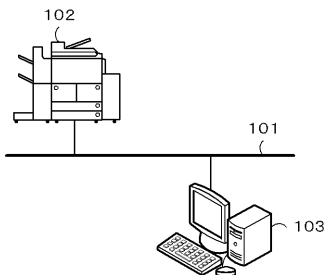
本発明は上記実施例に限定されるものではなく、本発明の趣旨に基づき種々の変形(各実施例の有機的な組合せを含む)が可能であり、それらを本発明の範囲から除外するものではない。即ち、上述した各実施例及びその変形例を組み合わせた構成も全て本発明に含まれるものである。

#### 【符号の説明】

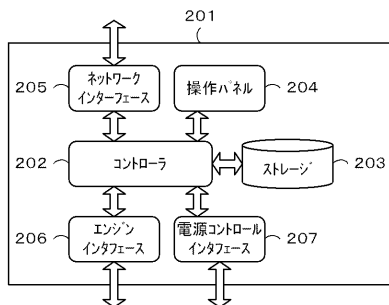
#### 【0245】

- 101 ネットワーク
- 102 画像形成装置(MFP)
- 103 情報処理装置

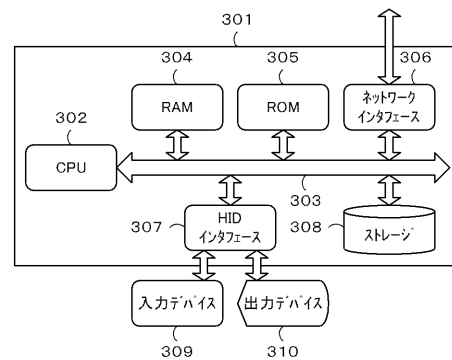
【図1】



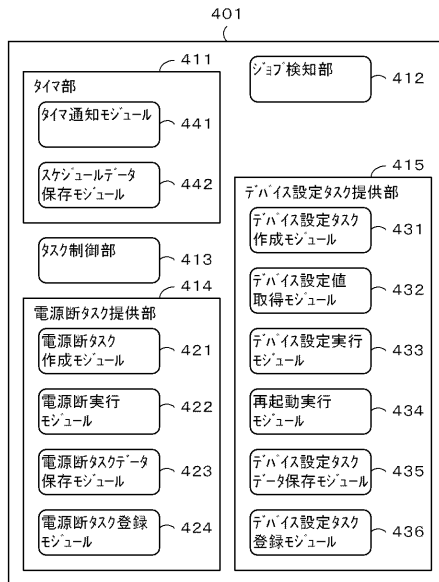
【図2】



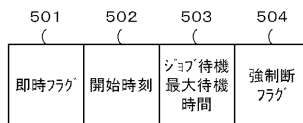
【図3】



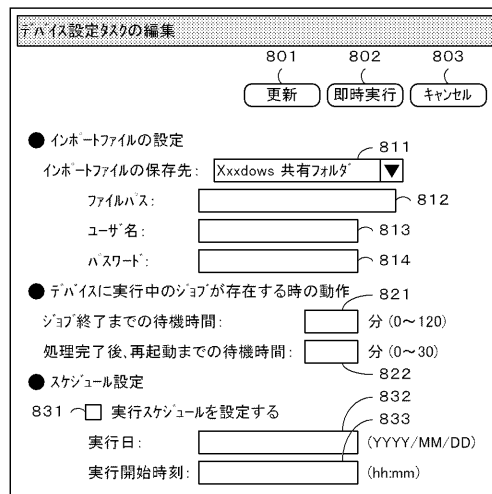
【図 4】



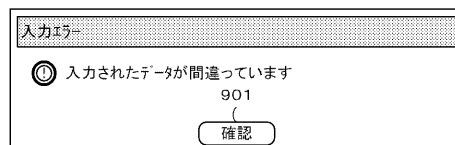
【図 5】



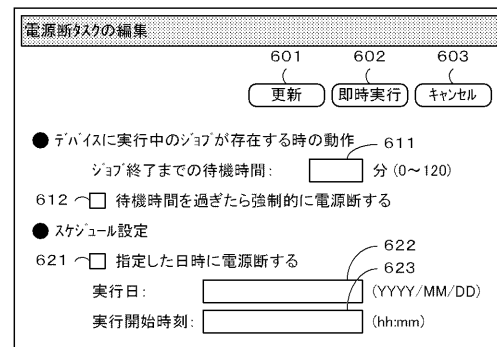
【図 8】



【図 9】



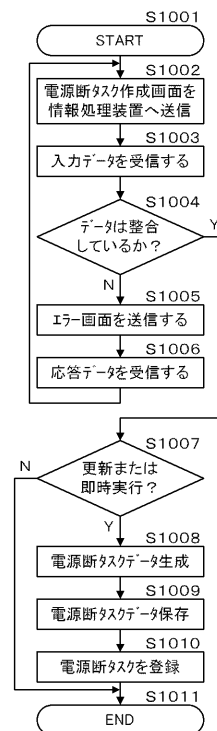
【図 6】



【図 7】



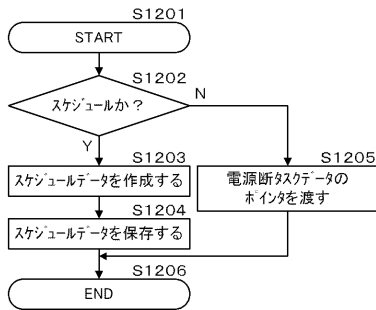
【図 10】



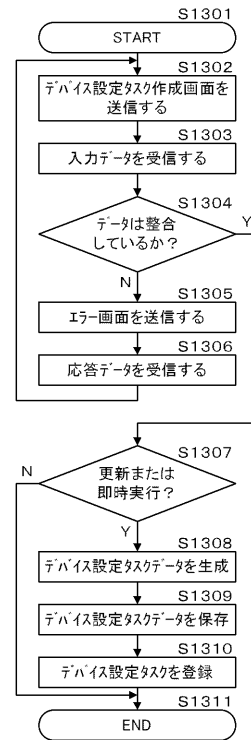
【図 11】



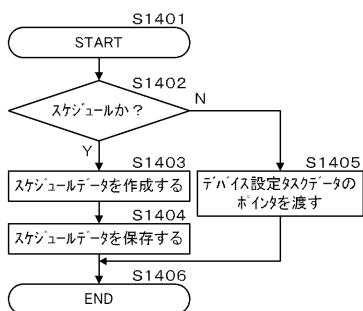
【図 12】



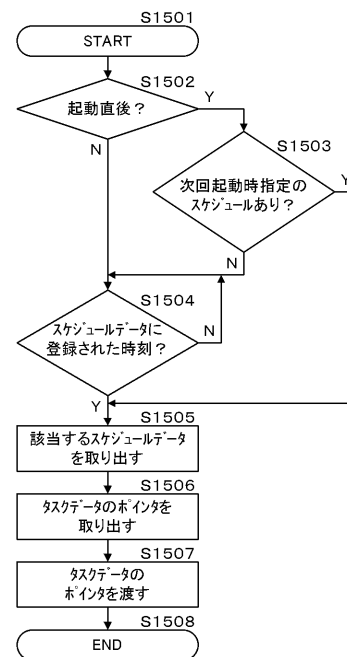
【図 13】



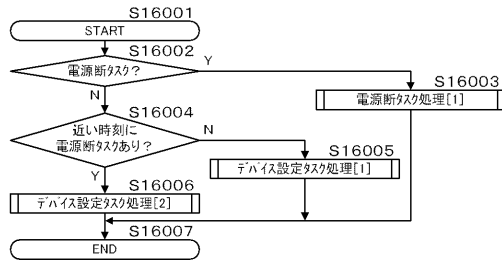
【図 14】



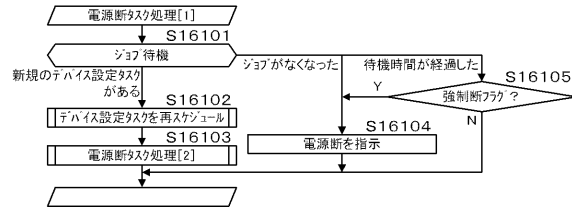
【図 15】



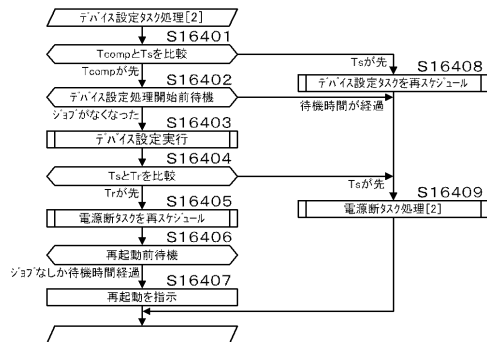
【図 16 A】



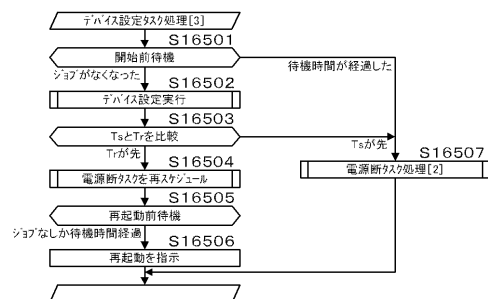
【図 16 B】



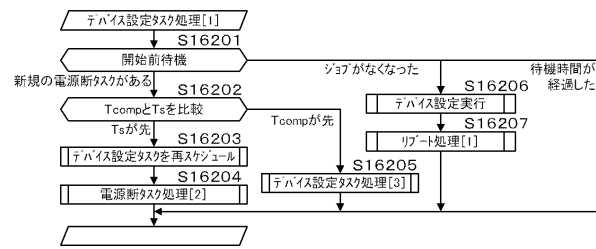
【図 16 E】



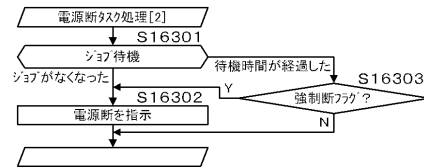
【図 16 F】



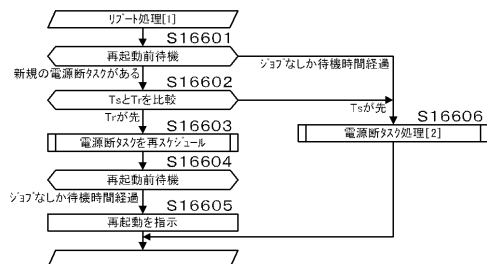
【図 16 C】



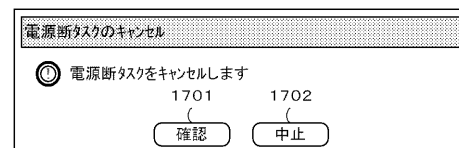
【図 16 D】



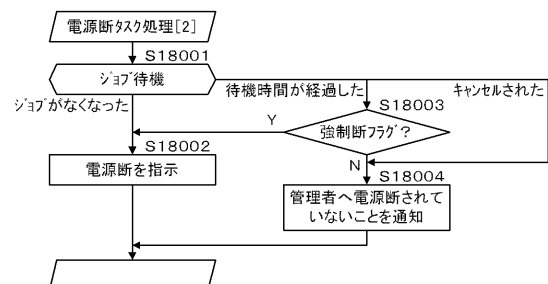
【図 16 G】



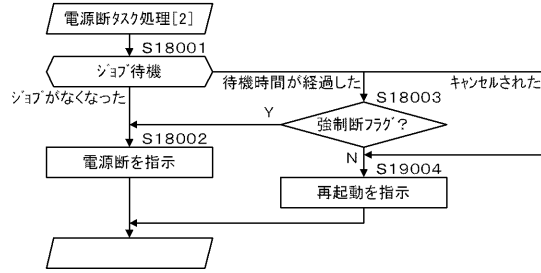
【図 17】



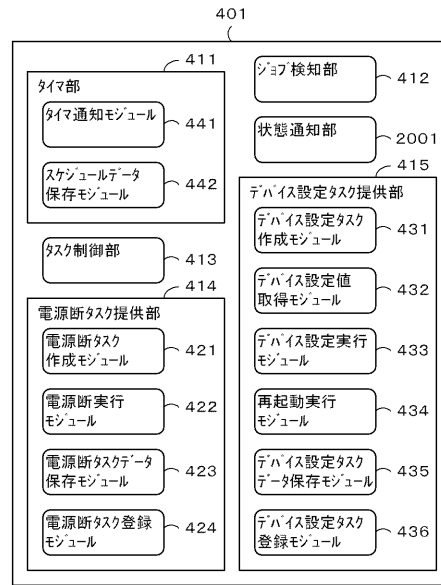
【図 18】



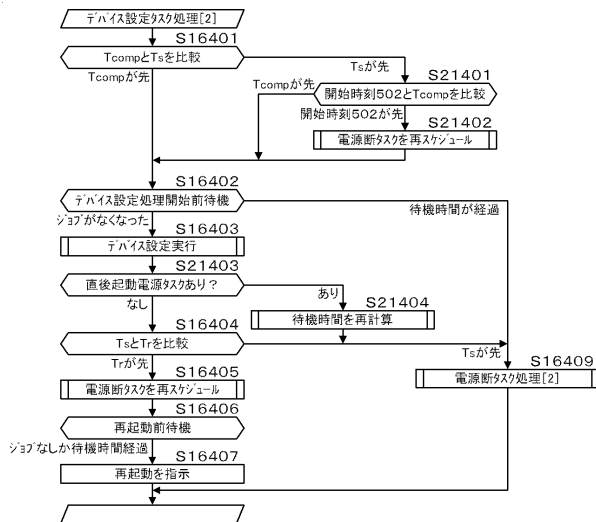
【図 19】



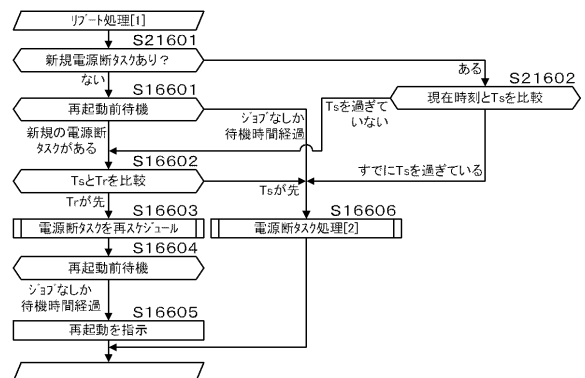
【図 20】



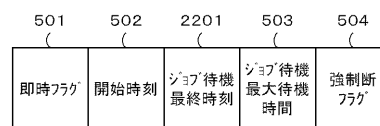
【図 21 A】



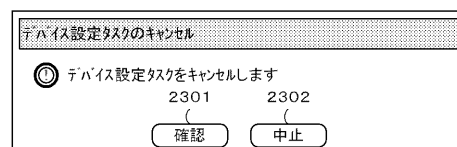
【図 21 B】



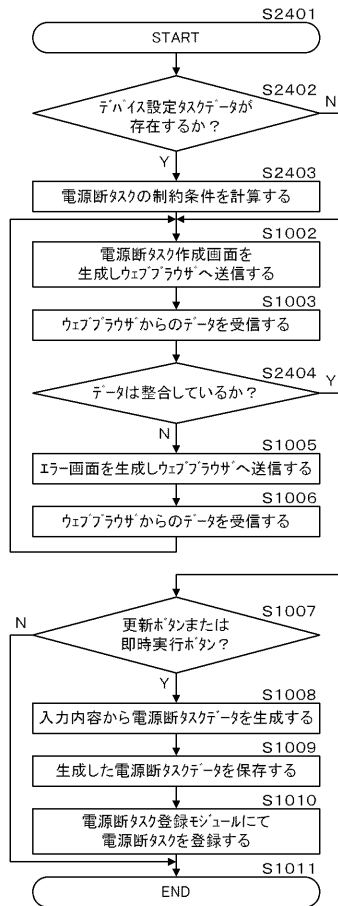
【図 22】



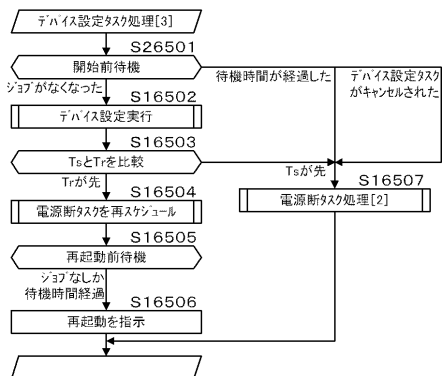
【図 23】



【図 24】



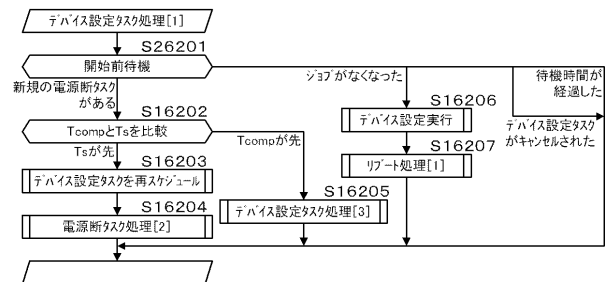
【図 26 B】



【図 25】

電源断タスクの編集		
2501	601 更新	602 即時実行
	603 キャンセル	
<p>すでにデバイス設定タスクが存在するため、ジョブ終了までの待機時間を加えて電源断がYYYY/MM/DD hh:mmまでに実行されることを強制する電源断タスクは作成できません。制限に抵触しない時刻と待機時間を入力するかデバイス設定タスクをキャンセルしてから再度作成してください。</p>		
<p>● デバイスに実行中のジョブが存在する時の動作 611</p> <p>ジョブ終了までの待機時間: <input type="text"/> 分 (0~120)</p>		
<p>612 <input type="checkbox"/> 待機時間をすぎたら強制的に電源断する</p>		
<p>● スケジュール設定</p> <p>621 <input type="checkbox"/> 指定した日時に電源断する</p>		
実行日:	<input type="text"/>	622 (YYYY/MM/DD)
実行開始時刻:	<input type="text"/>	623 (hh:mm)

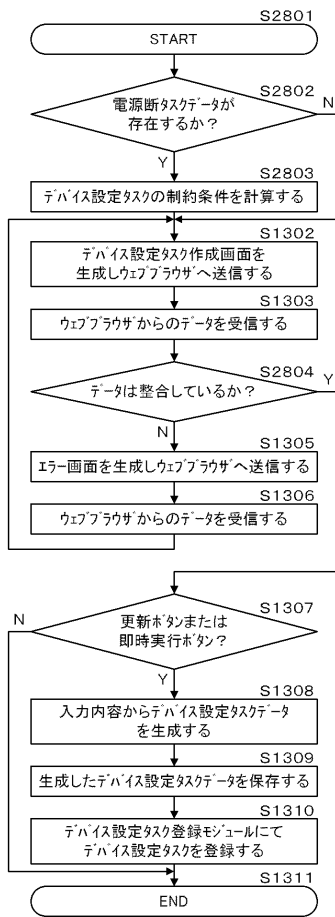
【図 26 A】



【図 27】

デバイス設定タスクの編集		
2701	801 更新	802 即時実行
	803 キャンセル	
<p>すでに電源断タスクが存在するため、ジョブ終了までの待機時間を加えてデバイス設定がYYYY/MM/DD hh:mmまでに実行されることを強制するデバイス設定タスクは作成できません。制限に抵触しない時刻と待機時間を入力するか電源断タスクをキャンセルしてから再度作成してください。</p>		
<p>● インポートファイルの設定</p> <p>インポートファイルの保存先: <input type="text"/> Xxxdows 共有フォルダ 811</p>		
ファイルパス:	<input type="text"/>	812
ユーザ名:	<input type="text"/>	813
パスワード:	<input type="text"/>	814
<p>● デバイスに実行中のジョブが存在する時の動作 821</p> <p>ジョブ終了までの待機時間: <input type="text"/> 分 (0~120)</p>		
<p>処理完了後、再起動までの待機時間: <input type="text"/> 分 (0~30)</p>		
<p>● スケジュール設定</p> <p>831 <input type="checkbox"/> 実行スケジュールを設定する</p>		
実行日:	<input type="text"/>	832 (YYYY/MM/DD)
実行開始時刻:	<input type="text"/>	833 (hh:mm)

【図 28】



---

フロントページの続き

(58)調査した分野(Int.Cl. , DB名)

H 0 4 N 1 / 0 0

B 4 1 J 2 9 / 3 8

G 0 3 G 2 1 / 0 0

G 0 6 F 3 / 1 2