

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第5656519号  
(P5656519)

(45) 発行日 平成27年1月21日(2015.1.21)

(24) 登録日 平成26年12月5日(2014.12.5)

(51) Int.Cl.

H04N 1/00 (2006.01)

F 1

H04N 1/00

C

請求項の数 12 (全 25 頁)

(21) 出願番号 特願2010-196527 (P2010-196527)  
 (22) 出願日 平成22年9月2日 (2010.9.2)  
 (65) 公開番号 特開2012-54803 (P2012-54803A)  
 (43) 公開日 平成24年3月15日 (2012.3.15)  
 審査請求日 平成25年8月30日 (2013.8.30)

(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  
 (72) 発明者 金松 伸一  
 東京都大田区下丸子3丁目30番2号 キ  
 ャノン株式会社内  
 審査官 鈴木 明

最終頁に続く

(54) 【発明の名称】情報処理装置、情報処理装置の制御方法、及び、プログラム

## (57) 【特許請求の範囲】

## 【請求項 1】

ファームウェアにより動作する情報処理装置であって、  
 前記ファームウェアの動作中のログを記憶する第1の記憶手段と、  
 予め設定された事象の発生に対応して、前記第1の記憶手段に記憶されているログを第  
 2の記憶手段に退避させる退避手段と、  
前記退避手段により前記第1の記憶手段から前記第2の記憶手段にログの退避が行われた場合、その旨を通知する第1の通知手段と、

操作者により設定された文字列を読み込む読込手段と、

前記退避手段により前記第2の記憶手段に退避されたログに、前記操作者により設定さ  
 れた文字列が含まれるか否かを検査するログ検査手段と、 10

前記ログ検査手段による検査結果を前記操作者に通知する第2の通知手段と、  
 を有することを特徴とする情報処理装置。

## 【請求項 2】

ファームウェアにより動作する情報処理装置であって、

前記ファームウェアの動作中のログを記憶する第1の記憶手段と、

予め設定された事象の発生に対応して、前記第1の記憶手段に記憶されているログを第  
 2の記憶手段に退避させる退避手段と、

操作者により設定された文字列をUSBメモリから読み込む読込手段と、

前記退避手段により前記第2の記憶手段に退避されたログに、前記操作者により設定さ

10

20

れた文字列が含まれるか否かを検査するログ検査手段と、  
前記ログ検査手段による検査結果を前記操作者に通知する通知手段と、  
を有することを特徴とする情報処理装置。

**【請求項 3】**

前記第1の記憶手段は、前記情報処理装置のメインメモリの一部領域に設けられたリンクバッファであることを特徴とする請求項1又は2に記載の情報処理装置。

**【請求項 4】**

前記第2の記憶手段に退避したログを前記情報処理装置と通信可能な外部装置に転送する転送手段を有することを特徴とする請求項1乃至3のいずれか1項に記載の情報処理装置。

10

**【請求項 5】**

前記外部装置は、前記情報処理装置に着脱可能に接続される記憶装置であることを特徴とする請求項4に記載の情報処理装置。

**【請求項 6】**

前記外部装置は、ネットワークを介して前記情報処理装置と通信可能な装置であることを特徴とする請求項4に記載の情報処理装置。

**【請求項 7】**

前記情報処理装置は、ファームウェアの動作により制御される複数の機能を有するものであり、

前記予め設定された事象を設定するための設定手段を有し、

20

前記設定手段は、前記予め設定された事象を前記機能ごとにそれぞれ設定することを特徴とする請求項1乃至6のいずれか1項に記載の情報処理装置。

**【請求項 8】**

前記情報処理装置の操作者による前記いずれの機能についてログを退避させるかの選択、及び、予め用意されている複数のモードのいずれのモードでログを退避させるかの選択を受け付ける受付手段と、

前記機能ごと及び前記モードごとに、前記予め設定された事象を含むログ設定情報を記憶するログ設定情報記憶手段と、を有し、

前記設定手段は、前記受付手段で受け付けた前記機能の選択及び前記モードの選択に対応するログ設定情報を前記ログ設定情報記憶手段から読み出して、前記予め設定された事象を設定することを特徴とする請求項7に記載の情報処理装置。

30

**【請求項 9】**

前記設定手段は、前記ログ設定情報を前記ログ設定情報記憶手段又は外部装置から読み出すことを特徴とする請求項8に記載の情報処理装置。

**【請求項 10】**

ファームウェアにより動作する情報処理装置の制御方法であって、

退避手段が、予め設定された事象の発生に対応して、第1の記憶手段に記憶されている前記ファームウェアの動作中のログを第2の記憶手段に退避させる退避ステップと、

第1の通知手段が、前記退避ステップにより前記第1の記憶手段から前記第2の記憶手段にログの退避が行われた場合、その旨を通知する第1の通知ステップと、

40

読み手段が、操作者により設定された文字列を読み込む読み込みステップと、

ログ検査手段が、前記退避手段により前記第2の記憶手段に退避されたログに、前記操作者により設定された文字列が含まれるか否かを検査するログ検査ステップと、

第2の通知手段が、前記ログ検査ステップによる検査結果を前記操作者に通知する第2の通知ステップと、

を有することを特徴とする情報処理装置の制御方法。

**【請求項 11】**

ファームウェアにより動作する情報処理装置の制御方法であって、

退避手段が、予め設定された事象の発生に対応して、第1の記憶手段に記憶されている前記ファームウェアの動作中のログを第2の記憶手段に退避させる退避ステップと、

50

読み手段が、操作者により設定された文字列をＵＳＢメモリから読み込む読み込みステップと、

ログ検査手段が、前記退避手段により前記第2の記憶手段に退避されたログに、前記操作者により設定された文字列が含まれるか否かを検査するログ検査ステップと、

通知手段が、前記ログ検査ステップによる検査結果を前記操作者に通知する通知ステップと、

を有することを特徴とする情報処理装置の制御方法。

【請求項12】

ファームウェアにより動作する情報処理装置を、請求項1乃至9のいずれか1項に記載された情報処理装置の手段として機能させるためのプログラム。 10

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、情報処理装置、情報処理装置の制御方法、及び、プログラムに関する。

【背景技術】

【0002】

近年の画像処理装置、特に複数の機能を備えたマルチファンクション機器（以下、MFPと略する）においては、機能の多様化、デバイスの高性能化に伴い、その制御を行うファームウェアの規模、複雑度が増大しつづけている。そのためファームウェア開発、デバッグにおける効果的な調査、分析手段が求められている。 20

【0003】

従来の組み込み装置向けのファームウェア開発においては、採用されるCPUに対応してハードウェアレベルによるソフトデバッグ支援ツールが提供されることが多かった。このようなハードウェアレベルによるデバッグ支援機能では、ソフト的には実現できない強力なデバッグ機能を具備しており、ソフトウェア開発における必須ツールとなっていた。

【0004】

例えば、ソフトウェアバグによる不正なメモリ領域アクセスの検出や、多様な条件でのプログラム実行停止などは、ハード支援ならではの強力なデバッグ機能であり、広くファームウェア開発で使用してきた。

【0005】

しかし、近年のMFPのコントローラでは、ハードウェアによるデバッグ支援装置が提供されない汎用CPUを採用することも増えている。そのような場合、主にソフトウェア中に埋め込んだデバッグメッセージをデバッグログとして出力し、そのログ内容を検証することで動作確認や不具合調査に使うことが広く行われている。 30

【0006】

デバッグログによるデバッグは、ハード支援に機能が劣るケースがある一方で、市場のユーザ先でも取得が可能であり、製品出荷後のファーム不具合調査でも利用することができる。そのため、MFPの市場不具合調査、分析にはまず市場から不具合動作を含むデバッグログを回収することが広く行われている。また同時に、ファームウェアの不具合発生時にその不具合要因の切り分けあるいは特定に利用できると、不具合への対応の迅速化が期待できる。 40

【0007】

そのため例えば、特許文献1のように、MFPの動作異常時のログを、異常が発生したデバイスのIDを付加して記録し、不具合原因の特定に役立てる技術が提案されている。

【先行技術文献】

【特許文献】

【0008】

【特許文献1】特開2008-299565号公報

【発明の概要】

【発明が解決しようとする課題】

50

**【0009】**

しかし、MFPのデバッグログには以下の課題がある。

MFPのようなリアルタイム制御を含むファームウェア動作のデバッグログ出力では、ミリ秒又はマイクロ秒オーダーの動作分析が必要である。そのためデバッグログ出力は、高速なメインメモリの一部領域をリングバッファとして行い、後からHDDにファイル化する構成をとることが多い。

**【0010】**

メインメモリは、高速な一方でデバッグログに割り当て可能な領域サイズは小さく、時間的に数秒～数十秒程度しかログ記録できない場合もある。そのため適切なタイミングでメインメモリからHDDなどの二次記憶装置にデバッグログをファイル化しないと、不具合調査に必要な動作区間の情報自体が失われてしまう。10

**【0011】**

この制約のため、従来では、取得したデバッグログに調査対象としたい動作が含まれず、分析自体ができない場合があった（第1の課題）。

さらに、MFPの機能としてデバッグログの制御機能やステータス確認機能が正式には提供されず、あくまで開発者向けの機能を限定的に運用するケースが多い。そのため、実際にMFPのログ取得を試みるサービスマンやユーザにとっては、前述の取得タイミングの問題と合わせてデバッグログの取り扱いが難しく、判りにくいという課題があった（第2の課題）。

**【0012】**

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、情報処理装置の制御ファームウェアにおける所望のデバッグログの取得の確率を向上し、ファームウェアの不具合調査の効率を高めることができる仕組みを提供することである。20

**【課題を解決するための手段】****【0013】**

本発明は、ファームウェアにより動作する情報処理装置であって、前記ファームウェアの動作中のログを記憶する第1の記憶手段と、予め設定された事象の発生に対応して、前記第1の記憶手段に記憶されているログを第2の記憶手段に退避させる退避手段と、前記退避手段により前記第1の記憶手段から前記第2の記憶手段にログの退避が行われた場合、その旨を通知する第1の通知手段と、操作者により設定された文字列を読み込む読込手段と、前記退避手段により前記第2の記憶手段に退避されたログに、前記操作者により設定された文字列が含まれるか否かを検査するログ検査手段と、前記ログ検査手段による検査結果を前記操作者に通知する第2の通知手段と、を有することを特徴とする。30

**【発明の効果】****【0014】**

本発明によれば、情報処理装置の制御ファームウェアにおける所望のデバッグログの取得の確率を向上し、ファームウェアの不具合調査の効率を高めることができる。

**【図面の簡単な説明】****【0015】**

【図1】本発明のMFPの全体構成を示す概要図である。

【図2】本発明のMFPのコントローラユニットのハードウェア構成の一例を示すプロック図である。

【図3】本発明のMFPのコントローラソフトウェアの構成図である。

【図4】本実施例におけるデバッグログ出力用API定義および呼び出し例を示す図である。

【図5】本実施例におけるデバッグログのデフォルト出力内容の一例を示す図である。

【図6】本実施例のデバッグログ動作設定ファイルの反映フローを示す図である。

【図7】本実施例のデバッグログ出力処理の一例を示すフローチャートである。

【図8】本実施例のデバッグログファイルの蓄積、回収フローを示す図である。

【図9】本実施例のログ設定ファイルのデータ例を示す図である。50

【図10】本実施例のデバッグログのIDによる絞込み出力内容の一例を示す図である。

【図11】本実施例のデバッグログのレベルによる絞込み出力内容の一例を示す図である。

【図12】本実施例のデバッグログのID及びレベルでの絞りこみ出力内容の一例を示す図である。

【図13】本発明のMFPのデバッグログステータス確認画面の一例を示す図である。

【図14】本実施例のデバッグログステータス画面のステータス表示処理の一例を示すフローチャートである。

【図15】本実施例でUSBメモリにログ回収する処理の一例を示すフローチャートである。 10

【図16】本発明のMFPのデバッグログの動作設定画面の一例を示す図である。

【図17】通常のコピー機能画面の標準画面の一例を示す図である。

【図18】通常のコピー機能画面の標準画面の一例を示す図である。

【図19】本実施例のデバッグログ自動退避処理の一例を示すフローチャートである。

【図20】本実施例でFTPサーバにログ回収する処理の一例を示すフローチャートである。

#### 【発明を実施するための形態】

##### 【0016】

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】 20

##### 【0017】

図1は、本発明の情報処理装置の一実施例を示す画像処理装置を適用したMFPの全体構成を示す概要図である。

図1において、100は本発明の画像処理装置を適用したMFP (Multifunction Peripheral) である。以下、MFP100の画像形成動作について説明する。

MFP100は、1Dカラー系画像処理装置(1つの感光ドラムでカラー画像形成を行う画像処理装置)であり、操作部210、スキャナエンジン201、プリンタエンジン202、さらにコントローラユニット200を有する。コントローラユニット200については後述の図2で説明する。なお、プリンタエンジン202は、レーザ露光部10、作像部20、定着部30、給紙/搬送部40を有する。 30

##### 【0018】

操作部210は、ユーザからの操作や装置全体のステータス表示などを行なう入力キー やタッチパネル付き表示画面を有し、画像処理装置とユーザとのユーザインターフェース機能を提供する。

##### 【0019】

スキャナエンジン201は、原稿台に置かれた原稿に対して、照明を当てて原稿画像を光学的に読み取り、その像を電気信号に変換して画像データを作成する。

プリンタエンジン202のレーザ露光部10は、画像データ(例えば、スキャナエンジン201により生成)に応じて変調されたレーザ光等の光線を等角速度で回転する回転多面鏡(ポリゴンミラー)に入射させ、反射走査光として感光ドラム21に照射する。 40

##### 【0020】

作像部20は、感光ドラム21を回転駆動し、帯電器によって帯電させ、レーザ露光部10によって感光ドラム21上に形成された潜像をトナーによって現像化し、そのトナー像をシートに転写する。その際に転写されずに感光ドラム21上に残った微小トナーを回収するといった一連の電子写真プロセスを実行して作像する。その際、シートが転写ドラム22の所定位置に巻きつき、4回転する間に、マゼンタ(M)、シアン(C)、イエロー(Y)、ブラック(K)のトナーを持つそれぞれの現像ユニット(現像ステーション)23~26が入れ替わりで順次前述の電子写真プロセスを繰り返し実行する。4回転の後、4色のフルカラートナー像を転写されたシートは、転写ドラム22を離れ、定着部30へ搬送される。 50

## 【0021】

定着部30は、ローラやベルトの組み合わせによって構成され、ハロゲンヒータなどの熱源を内蔵し、前記作像部によってトナー像が転写されたシート上のトナーを、熱と圧力によって溶解、定着させる。

## 【0022】

給紙／搬送部40は、シートカセットやペーパーデッキに代表されるシート収納庫を一つ以上持つており、前記プリンタ制御部の指示に応じてシート収納庫に収納された複数のシートの中から一枚分離し、作像部20・定着部30へ搬送する。シートは作像部20の転写ドラム22に巻きつけられ、4回転した後に定着部30へ搬送される。4回転する間に前述のYMC各色のトナー像がシートに転写される。また、シートの両面に画像形成する場合は、定着部を通過したシートを再度、作像部20へ搬送する搬送経路を通りよう10に制御する。

## 【0023】

上記各機能部はコントローラユニット200で実行されるコントローラウェアによって統合制御されており、それぞれコントローラソフトウェアと相互に通信を行ないつつ画像処理装置機能動作を実現するものである。コントローラソフトウェアについては図3で後述する。なお、上記のとおり図1のMFPの制御においてはコントローラソフトウェアは紙搬送制御、画像データ生成など実時間制約の強いリアルタイム処理を行う必要がある。その内部処理の時間粒度は、μ秒のオーダーであることも多く、コントローラソフトウェアが出力するデバッグログの出力頻度は通常のパーソナルコンピュータ（PC）やサーバ20上のアプリケーションの数百～数万倍となる。

## 【0024】

<コントローラハードの構成>

図2は、コントローラユニット200のハードウェア構成の一例を示すブロック図である。

## 【0025】

図2に示すように、コントローラユニット200は、画像入力デバイスであるスキャナエンジン201や画像出力デバイスであるプリンタエンジン202と接続し、画像データの読み取りやプリント出力のための制御を行う。また、コントローラユニット200は、LAN50と接続することで、外部装置（例えば、FTPサーバ250）と画像情報やデバイス情報を入出力するための制御を行う。30

## 【0026】

コントローラユニット200において、MainCPU205は、主に装置全体のシステム的な制御とアプリケーションプログラムを制御するための中央処理装置である。RAM206は、MainCPU205が動作するためのシステムワークメモリであり、MainCPU205が出力するデバッグログを保持する領域も確保されている。

## 【0027】

SubCPU218は、主にスキャナエンジン201、プリンタエンジン202のリアルタイム制御、及び画像処理の制御をつかさどり、MainCPU205と連携してコピー機能、PDLプリント機能を実現する。なお、PDLは、Page Description Languageの略称であり、ページ記述言語を示す。即ち、PDLプリント機能とは、ページ記述言語で記述されたプリントデータに基づいて印刷を行う機能を示す。40

## 【0028】

RAM219は、SubCPU218用のメインメモリであり、SubCPU218が出力するデバッグログ出力領域もここに割り当てられている。

BootROM207には、コントローラの電源がON後、MainCPU205を起動するために最初に実行されるシステムのブートプログラム（図3のブートローダー370）が格納されている。このブートプログラムにより、BootROM207内のオペレーティングシステム（図3の380）や各種ファームウェアが起動される。ここで、電源Onで起動されるのはMainCPU205のみである。SubCPU218の起動は、50

MainCPU205のファームウェアがSubCPU218用のファームウェアをBootROM207からロードし、起動させることで行われる。

#### 【0029】

ハードディスクドライブ(HDD)208は、各種処理のためのシステムソフトウェア及び入力された画像データ、及び本発明のデバッグルогが参照するログ設定ファイルA1(図6)、自動退避によって永続化されたログファイルB(図6)を保持する。なお、HDD208の代わりに、SSD(Solid State Drive)を備える構成であってもよい。

#### 【0030】

操作部I/F209は、画像データ等を表示可能なタッチパネル付き表示画面を有する操作部210に対するインターフェース部であり、操作部210に対して操作画面データを出力する。また、操作部I/F209は、操作部210から操作者が入力した情報をMainCPU205に伝える役割をする。ネットワークインターフェース(ネットワークI/F)211は、例えばLANカード等で実現され、LAN50に接続して外部装置(例えば、FTPサーバ250)との間で情報の入出力をを行う。USB I/F230は、USBデバイス(例えば、USBメモリ240)との間で情報の入出力をを行う。なお、USBメモリ240の代わりに、USBハードディスク等のMFP100と着脱可能な他の記憶装置であってもよい。もちろん、インターフェースはUSBインターフェースでなくとも、MFP100と通信可能なインターフェースであればどのようなインターフェースであってもよい。

#### 【0031】

イメージバスインターフェース(イメージバスI/F)214は、システムバス213と画像データを高速で転送する画像バス215とを接続するためのインターフェースであり、データ構造を変換するバスブリッジである。画像バス215上には、ラスタイメージプロセッサ(RIP)216、デバイスインターフェース(デバイスI/F)217が接続される。

#### 【0032】

RIP216は、ページ記述言語(PDL)コードや後述するベクトルデータをイメージに展開する。デバイスI/F部217は、スキャナエンジン201やプリンタエンジン202とコントローラユニット200とを物理的に接続し、画像データの同期系/非同期系の変換を行う。

#### 【0033】

また、画像処理ボード203は、スキャナエンジン201から読み込んだ画像やプリンタエンジン202に出力しようとしている画像に応じた補正、解像度変換等の処理を行う。画像処理は、スキャナエンジン201、プリンタエンジン202の速度に対応したリアルタイム処理が必要なため、画像処理ボード203の高速な専用処理ハードウェアによって処理実行される。

#### 【0034】

FAXデバイス260は、ファクシミリプロトコルによって通信を行うためのデバイスである。

図2中のFTPサーバ250とUSBメモリ240は、本実施例のMFP100のコントローラハードウェアに含まれるものではないが、後述の実施例中でデバッグルогの回収に使用する他の外部装置として使用されるものである。

#### 【0035】

<コントローラソフトウェア構成>

図3は、MFP100のコントローラユニット200で実行されて画像処理装置全体の制御を行なうコントローラソフトウェアの構成図である。

#### 【0036】

図3において、300はコントローラソフトウェアである。コントローラソフトウェア300は、MainCPU205で実行されるMain側と、SubCPU218で実行されるSub側から構成されるが、本実施例で説明するデバッグルогの動作は主にMain

10

20

30

40

50

n C P U 2 0 5 に関する。そのため S u b C P U 2 1 8 側の詳細説明は省略する。

#### 【 0 0 3 7 】

コントローラソフトウェア 3 0 0 は、システム全体制御とアプリケーション層の制御とともに、S u b C P U 2 1 8 と連携してスキャナエンジン 2 0 1、プリンタエンジン 2 0 2、画像処理などのリアルタイム処理も行う。また、コントローラソフトウェア 3 0 0 は、操作部 2 1 0 から入力された動作指示にしたがって M F P 全体としての機能動作を実現する。ブートローダー 3 7 0 は、コントローラユニット 2 0 0 のオペレーティングシステム 3 8 0 を起動するためのシステム読み込みプログラムである。オペレーティングシステム 3 8 0 はコントローラソフトウェア 3 0 0 全体が動作するための実行環境およびデバイスアクセス機能を提供するシステムソフトウェアである。

10

#### 【 0 0 3 8 】

U I アプリケーション 3 0 1 は、操作部 2 1 0 の画面表示及び入力受付を行うソフトウェアモジュールである。ネットワークモジュール 3 0 2 は、ネットワーク I / F 2 1 1 のドライバ及びプロトコル層の制御を行い、ネット - ワークを介した通信機能を提供する。

#### 【 0 0 3 9 】

U S B 管理モジュールは、U S B I / F 2 3 0 を管理し、U S B デバイスの装着や取り外しを監視する。スキャンジョブ 3 1 0 、コピージョブ 3 2 0 、プリントジョブ 3 3 0 、F A X ジョブ 3 6 0 は、それぞれ対応する M F P 機能を実現するソフトウェアモジュールである。

20

#### 【 0 0 4 0 】

コントローラ A P I 3 0 3 は、アプリケーション層からの動作指示を受け付けるソフト A P I 群で構成され、アプリケーションが所定の手順で本 A P I 群を呼び出すことで M F P 動作が実行される。

#### 【 0 0 4 1 】

電力管理モジュール 3 8 3 は、M F P 全体の電力制御を行い、電源 O F F 時のシステム電源遮断処理や、M F P がアイドル状態の場合に省電力モードにシステム状態を変更する制御を行う。エラーマネージャ 3 8 5 は、M F P 全体の各種事象を監視しており、例えばコントローラソフトウェア 3 0 0 内で発生したソフト例外、あるいはサービスマンコールのような致命的エラー、プリンタエンジン 2 0 2 でのジャムなどを捕捉するモジュールである。

30

#### 【 0 0 4 2 】

< デバッグログ管理モジュール構成 >

デバッグログ管理モジュール 3 8 2 は、コントローラソフトウェア 3 0 0 が動作中に出力するデバッグログの出力、退避、転送など一連のデバッグログ機能を実現する。

#### 【 0 0 4 3 】

次に、デバッグログ管理モジュール 3 8 2 が提供するデバッグログ出力 I F 機能について図 4 を用いて説明する。

図 4 は、本実施例におけるデバッグログ出力用 A P I 定義及び呼び出し例を示す図である。

デバッグログ出力 I F 8 3 0 は、コントローラソフトウェア 3 0 0 の開発者向けに提供されている。

40

なお、本実施例では、ログ I D 定義 8 2 0 に示すように、各モジュール用にログ I D が定義されているものとする。例えば、U I アプリケーション 3 0 1 用には「U I 0 0」から始まる一連の I D が、F A X ジョブ 3 6 0 用には「F A X 0 0」からの一連の I D が、アプリ用には「A P P 0 0」からの一連の I D が割り当てられる。

#### 【 0 0 4 4 】

また、本実施例では、ログ出力レベル定義 8 1 0 に示すように、出力レベルが定義されているものとする。なお、出力レベルは、ログ出力のエラー程度に応じて設定される。

8 4 0 は、デバッグログ出力 I F 8 3 0 を用いたデバッグログ出力呼出例である。この例では、ログ I D として F A X 0 0 、出力レベルとして W A R N 、文字列として "Waitign

50

IMG-MEM 1000ms retry"が指定されて、デバッグログ出力が呼び出されている。

#### 【0045】

本実施例のコントローラソフトウェア300には、開発者により、840のようなデバッグログ出力呼出が、プログラムコード内に埋め込まれている。開発者は、840のようなデバッグログ出力呼出により出力されるデバッグログの内容を検証することで、コントローラソフトウェア300の動作確認や不具合調査を行うことができる。

#### 【0046】

図5は、デバッグログ出力の一例を示す図である。なお、本発明の効果を端的に説明するため、本実施例ではRAM206に割り当てるデバッグログ用領域（図6に示すデバッグログ用RAM領域1040）をログ60行分とすることを前提として説明する。

10

#### 【0047】

図5ではRAM206のデバッグ領域60行分いっぱいにログ出力されている様子を示す。

次の61行目にあたるデバッグログ出力IF830の呼び出しが行われると、デバッグログ管理モジュール382は、RAM206領域をリングバッファとして最古の行を上書きしてログ記録を継続する。そのため、RAM206のデバッグログ領域には最近の60行分だけの出力が保持されており、それ以前のログは上書きされて失われていく構成である。

#### 【0048】

line910はログの行数、time920はデバッグログ出力が行われた時刻、TASK930はデバッグログ出力を行ったモジュールのタスク名を示す。GroupID940、LEVEL950、Message960はそれぞれデバッグログIF830の呼び出し時に入力されたログID、出力レベル、メッセージ本体の文字列である。例えば、図5の30行目にあたる970の出力は、図4の840のようにデバッグログ出力IFを呼び出した結果である。なお、行数910、時間920、タスク名930はデバッグログ管理モジュール382が自動的に付加する情報であり、開発者によるデバッグログ分析をし易くするためのものである。

20

#### 【0049】

図6は、デバッグログ管理モジュール382の内部構成と、その動作モードを制御するログ設定の反映フローを示す図である。

30

図7は、本実施例のデバッグログ出力処理の一例を示すフローチャートである。なお、このフローチャートの処理は、デバッグログ管理モジュール382により実行される（即ち、MainCPU205がBootROM207から読み出されたファームウェアを実行することにより実現される）。

#### 【0050】

デバッグログ出力モジュール1000は、デバッグログ出力IF830の処理実体である。デバッグログ出力モジュール1000は、ログ動作設定1010を参照しつつ、図7のフローチャートに従ってデバッグログ用RAM領域1040へのログ出力を行う。なお、ログ動作設定1010は、ログ設定ファイルA1やA2から読み込まれてものである。A1は、MFP100のHDD208（ログ設定情報記憶部）に標準保持されるログ設定ファイル（ログ設定情報）を示す。また、A2は、USBメモリ240等の外部装置に記憶されて後からMFP100に送り込まれるカスタムログの設定ファイル（ログ設定情報）を示す。なお、ログ設定ファイルA1やA2の具体例は、後述する図9に示す。

40

#### 【0051】

デバッグログ出力IF830が呼び出されると、デバッグログ出力モジュール1000は、図7のS1201において、入力されたログIDが有効（出力対象）かどうか、ログ動作設定1010内のログ出力条件を参照して判定する。なお、ここで参照するログ動作設定1010内のログ出力条件は、具体的には、後述する図9の1803、1813、1823、1833等の"ID:"の設定値が対応する。

#### 【0052】

50

そして、入力されたログIDが無効（出力対象でない）と判定した場合（S1201でNo）、デバッグログ出力モジュール1000は、ログ出力を行わずに処理を終了する。

一方、入力されたログIDが有効（出力対象）と判定した場合（S1201でYes）、デバッグログ出力モジュール1000は、S1202に処理を進める。

S1202では、デバッグログ出力モジュール1000は、入力された出力レベルが有効（出力対象）かどうか、ログ動作設定1010内の出力レベルの設定を参照してチェックする。なお、ここで参照するログ動作設定1010内の出力レベルの設定は、具体的には、後述する図9の1803、1813、1823、1833等の"Level:"の設定値が対応する。

#### 【0053】

10

そして、入力された出力レベルが無効（出力対象でない）と判定した場合（S1202でNo）、デバッグログ出力モジュール1000は、ログ出力を行わずに処理を終了する。

一方、入力された出力レベルが有効（出力対象）と判定した場合（S1202でYes）、デバッグログ出力モジュール1000は、S1203に処理を進める。

S1203では、デバッグログ出力モジュール1000は、入力されたメッセージに、タイムスタンプ、タスク名、ID、レベル等の他の情報を附加して、ログ動作設定1010内の出力先の設定で指定されている出力先に出力し、処理を終了する。なお、ログ動作設定1010内の出力先の設定とは、具体的には、後述する図9の1803、1813、1823、1833等の"output:"の設定値が対応する。例えば、図9の1803のように"output:RAM"と設定されている場合、図6の例では、デバッグログ用RAM領域1040となる。なお、デバッグログの出力先には高速なメインメモリを用いる必要があるため、通常では、デバッグログの出力先はデバッグログ用RAM領域1040となる。しかし、本実施例では、"output:"の設定値を変更することにより、デバッグログの出力先を変更することも可能である。

20

#### 【0054】

ログ設定読み込みモジュール1020は、HDD208又はUSBメモリ240に保持されているログ設定ファイルA1又はA2を読み込み、ログ動作設定1010に反映するモジュールである。

#### 【0055】

30

ここで、ログ設定ファイルA1は、MFP出荷時点での予め想定されるデバッグログ動作パターンを定義して標準状態でHDD208に保持されている設定ファイル群である。また、ログ設定ファイルA2は、想定されたログ設定ファイルA1のデバッグログ動作設定に含まれないカスタム設定ファイルであり、製品出荷後に想定外の条件でデバッグログ取得を行いたい場合に開発者が記述し作成するものである。

#### 【0056】

デバッグログ用RAM領域1040は、MFP100のメインメモリ(RAM206)の一部領域をデバッグ用に割り当てるリングバッファ(ログ記憶部)であり、本実施例ではデバッグログ60行分が確保されている。

#### 【0057】

40

続いて図8を用いてログの自動退避及び外部装置へのログファイル転送について説明する。

図8は、本実施例のデバッグログファイルの蓄積、回収フロー図である。

ログ自動退避モジュール1030は、ログ動作設定1010内のログ退避トリガ条件が発生すると、このタイミングでデバッグログ用RAM領域1040の内容を読み出し、HDD208の一時領域にファイルとして退避(自動退避)する。なお、ログ動作設定1010内のログ退避トリガ条件とは、具体的には、後述する図9の1804、1814、1824等の"trigger:"の設定値が対応する。また、MFP100内の各種事象は、エラーマネージャ385からログ自動退避モジュール1030に常に通知されており、実際にログ退避を行うかはログ自動退避モジュール1030が設定(ログ退避トリガ条件)に基づいて判断を行う。

50

**【0058】**

ログパターンチェックモジュール1060は、自動退避されたログファイルに、ログ動作設定1010内の期待パターンが含まれるかを検査するモジュールである。なお、ログ動作設定1010内の期待パターンとは、具体的には、後述する図9の1802、1812、1822、1832等で設定されている文字列（例えば、"Waitign IMG-MEM 1000ms retry"）が対応する。そして、上記自動退避されたログファイルに期待パターンが含まれている場合、ログ自動退避モジュール1030は、上記HDD208内の一時領域に自動退避されたログファイルをHDD208内のログ永続化領域（永続化記憶部）に退避し、期待ログ保存フラグをOnにする。なお、期待ログ保存フラグは、後述する図13の410に示される機能ごとに設けられたものであり、デバッグルог管理モジュール382内で管理され（例えば、RAM206又はHDD208内に保持され）、後述する図14のS1602で参照される。10

**【0059】**

ログ転送モジュール1050は、HDD208のログ永続化領域に退避されたログファイル群を外部装置（FTPサーバ250やUSBメモリ240）に転送するモジュールである。

**【0060】**

図9は、ログ設定ファイルA1又はA2の一例を示す図である。

ログ設定ファイルは、設定項目別の1つ以上のセクションから構成される。例えば、ログ設定ファイルは、期待ログパターン、ログ出力条件部、ログ退避トリガ、サーバ設定、上書きモード等のセクションが含まれる。20

**【0061】**

以下、本実施例のログ動作設定例、デバッグルог出力例では、「MFPが受信したFAXの処理に異常発生」の状況で異常原因調査のためのデバッグルог取得を想定して説明する。

**【0062】**

図9において、ログ設定ファイル1801は、期待ログパターン1802、ログ出力条件部1803、ログ退避トリガ1804等のセクションを有する。

ログ出力条件部1803は、出力先はRAM(output:RAM)、出力IDはFAX関連の全てのID(ID:FAX,all)、アプリの60番と63番(ID:App,60,63)、出力レベルは全レベル(Level:all)（絞り込み無し）の設定である。即ち、ログIDがFAX関連のID、アプリの60番、又は、アプリの63番のデバッグルог出力が呼び出された場合に、デバッグルогをデバッグルог用RAM領域1040に出力することが設定されている。なお、レベル(LEVEL)による絞込みは行わない設定となっている。30

**【0063】**

また、ログ退避トリガ1804は、FAX関連のエラー発生でRAM206に保持されているデバッグルог用RAM領域1040をファイル化する設定(trigger:FaxERR)である。即ち、FAX関連のエラーが発生した場合に、このタイミングでデバッグルог用RAM領域1040内のデバッグルогをHDD208の一時領域に退避することが設定されている。40

**【0064】**

また、期待ログパターン1802は、"Waitign IMG-MEM 1000ms retry"の設定である。即ち、上記一時領域に退避されたデバッグルог内に文字列"Waitign IMG-MEM 1000ms retry"が含まれている場合に、上記一時領域に退避されたログをHDD208内のログ永続化領域に退避することが設定されている。

**【0065】**

図10は、ログ設定ファイル1801の設定の元で出力されたデバッグルог内容例を示す図である。

図5に示したデフォルト出力内容では、コピー動作、ネットワークパケット等の動作等と混合してFAX動作のログが出力されており、FAXに注目したログ分析は困難である50

。一方で図10に示すように、FAXに絞ったログID出力の結果では、FAXに関連したアプリ及びFAXジョブのログのみが output されており、容量が限られたデバッグログ用RAM領域1040が有効利用でき、且つ、FAX動作分析も容易である。また、期待ログパターン1802で記述したメッセージもログ内に残っているため、ログパターンチェックモジュール1060は、期待したログが取得されたと判断し、ログ自動退避モジュール1030によるHDD208への退避が行われる。

#### 【0066】

次に、図9のログ設定ファイル1810について説明する。

期待ログパターン1812、ログ退避トリガ1814は、ログ設定ファイル1801の期待ログパターン1802、ログ退避トリガ1804と同一であるので説明は省略する。

ログ出力条件部1813は、出力先はRAM(output:RAM)、出力IDは全ログID(ID:all,all)、出力レベルは警告以上の出力(Level:WARN)の絞込みを行う設定例である。即ち、出力レベルが警告"WARN"以上のデバッグログ出力が呼び出された場合に、デバッグログをデバッグログ用RAM領域1040に出力することが設定されている。なお、ログIDによる絞込みは行わない。

#### 【0067】

図11は、ログ設定ファイル1810の設定で出力されたデバッグログ内容例を示す図である。

ログ設定ファイル1810では、ログID設定は「全ID(ID:all,all)」となっているため、ネットワーク、FAX、電力制御、プリントなどの各モジュールの出力がされている。出力レベルでの絞込みは、MFP全体の動作で発生しているエラーや警告を確認する目的では判りやすく、且つ、緊急度の高いログのみに絞り込むとともに、デバッグログ用RAM領域1040を節約することができる。

#### 【0068】

例えば、図11では、直接異常が発生しているFAX処理中のメモリ確保の待ちが確認できるとともに、同時に電力制御のIDでERRレベルのログが11行目で確認できる。そのためFAXと同時に電力関係の不具合が発生し、FAXに異常が生じたのではないかという推測や調査につなげることが可能となる。

#### 【0069】

次に、図9のログ設定ファイル1820について説明する。

期待ログパターン1822、ログ退避トリガ1824は、ログ設定ファイル1801の期待ログパターン1802、ログ退避トリガ1804と同一であるので説明は省略する。

ログ出力条件部1823は、出力先はRAM(output:RAM)、出力IDはFAX関連の全てのID(ID:FAX,all)、アプリの60番と63番(ID:App,60,63)、出力レベルは警告以上の出力(Level:WARN)の絞込みを行う設定例である。即ち、ログIDがFAX関連のID、アプリの60番、又はアプリの63番で、且つ、出力レベルが警告"WARN"以上のデバッグログ出力が呼び出された場合に、デバッグログをデバッグログ用RAM領域1040に出力することが設定されている。

#### 【0070】

図12は、ログ設定ファイル1820の設定で出力されたデバッグログ内容例を示す図である。

図12の例では、ログ設定ファイル1801によるID絞り込みによる出力である図10に比べて出力レベルが警告"WARN"未満のLOGレベルが出力されず、「FAX動作の警告"WARN"以上のログのみ」が取得されている。このように出力条件を絞り込むことで限られたデバッグログ用RAM領域1040がリングバッファとして古いログを上書きするタイミングを引き伸ばし、有効なログを記録する可能性を高めることができる。

#### 【0071】

次に、図9のログ設定ファイル1830について説明する。

ログ設定ファイル1830には、期待ログパターン1832、ログ出力条件部1833等に加えて、サーバ設定1834のセクションが記述されている。なお、期待ログパター

10

20

30

40

50

ン 1 8 3 2、ログ出力条件部 1 8 3 3 は、ログ設定ファイル 1 8 2 0 の期待ログパターン 1 8 2 2、ログ出力条件部 1 8 2 3 と同一であるので説明は省略する。

#### 【0072】

サーバ設定 1 8 3 4 は、サーバのネットワークアドレス ( SERVER:192.168.0.1 ) 、使用するアカウント ( USER:logger 、 PASS:xxxx ) を用いてサーバに接続を行う設定例である。なお、このサーバ設定 1 8 3 4 は、後述する図 2 0 の S 2 0 0 5 で用いられる。

#### 【0073】

<デバッグルог画面>

次に、デバッグルог管理モジュール 3 8 2 が提供する機能やステータスを設定、確認する U I 画面について図 1 3 を用いて説明する。

10

#### 【0074】

図 1 3 は、本 M F P のデバッグルог動作設定やログ自動取得の状態を表示、確認可能な操作部 2 1 0 のタッチパネル付き表示部に表示されるデバッグルогステータス画面の一例を示す図である。

#### 【0075】

図 1 3 に示すように、4 1 0 の列に示す M F P 機能ごとにログ設定が行われ、各機能ごとにログ自動取得設定 4 2 0 、取得モード 4 3 0 、取得状況 4 4 0 を表示する。なお、M F P 機能とは、例えば、コピー機能、プリント機能、B O X 機能、S E N D 機能、F A X 機能、電力機能、アプリ機能、全体機能等を示す。

20

#### 【0076】

図 1 3 では、コピー機能と F A X 機能でログ自動取得 4 2 0 が有効となっており、コピー機能では取得状況ボタン ( ステータスボタン ) 4 4 1 で自動取得は行われていないこと、F A X 機能では取得状況ボタン ( ステータスボタン ) 4 4 2 で自動取得が行われログが退避されていることが示されている。

#### 【0077】

また、ログ自動取得のタイミングが、取得モード 4 3 0 で設定されており、コピー機能は「 J A M 」、F A X 機能は「カスタム」の設定となっている。

以下、取得状況ボタン 4 4 1 及び 4 4 2 の表示処理を図 1 4 のフローチャートで説明する。

図 1 4 は、本実施例のデバッグルогステータス画面のステータス表示処理の一例を示すフローチャートである。なお、このフローチャートの処理は、U I アプリケーション 3 0 1 により実行される ( 即ち、M a i n C P U 2 0 5 が B o o t R O M 2 0 7 から読み出されたファームウェアを実行することにより実現される ) 。

30

#### 【0078】

操作部 2 1 0 からデバッグルогステータス画面 ( 図 1 3 ) を開く操作が行われると、U I アプリケーション 3 0 1 は、S 1 6 0 1 において、デバッグルогステータス画面を操作部 2 1 0 のタッチパネル付き表示部に表示する。

#### 【0079】

そして、U I アプリケーション 3 0 1 は、デバッグルогステータス画面内に表示する項目の収集を行い、その 1 つとして、S 1 6 0 2 において、機能ごとの期待ログ保存フラグを、デバッグルог管理モジュール 3 8 2 から取得する。なお、期待ログ保存フラグは、後述する図 1 9 の S 1 5 0 9 又は S 1 5 1 4 で設定されるものである。

40

#### 【0080】

次に、S 1 6 0 3 において、U I アプリケーション 3 0 1 は、上記 S 1 6 0 2 で取得した機能ごとの期待ログ保存フラグの状態に応じて、取得状況 4 4 0 に「あり」又は「なし」の表示を行う。詳細には、期待ログ保存フラグが O n の機能については、図 1 3 の 4 4 2 に示すように、取得状況 4 4 0 に「あり」の表示を行う。一方、期待ログ保存フラグが O f f の機能については、図 1 3 の 4 4 1 に示すように、取得状況 4 4 0 に「なし」の表示を行う。

#### 【0081】

50

以下、図13の画面の説明に戻る。

ユーザが、USBメモリ転送ボタン490を選択してから実行ボタン402を押下すると、デバッグログ管理モジュール382は、HDD208に自動退避されたデバッグログファイルをUSBメモリに転送実行する。ステータス表示ボタン491には、USBメモリへの転送動作状況が表示される。なお、USBメモリがMFP100に装着されている場合にのみ、USBメモリ転送ボタン490が操作可能になるように構成してもよい。

#### 【0082】

FTPサーバ転送ボタン492は、ログ転送先を所定のFTPサーバへ転送する場合に選択、実行され、USB転送と同様にステータスボタン493に処理状況が表示される。なお、図9のサーバ設定1834のようなFTPサーバの設定がなされている場合にのみ、FTPサーバ転送ボタン492が操作可能になるように構成してもよい。10

#### 【0083】

USBメモリ転送ボタンから行われる処理を、図15のフローチャートで説明する。

図15は、本実施例でUSBメモリにログ回収する処理のフローチャートである。なお、このフローチャートの処理は、UIアプリケーション301、USBIF管理モジュール390、デバッグログ管理モジュール382等により実行される（即ち、MainCP U205がBootROM207から読み出されたファームウェアを実行することにより実現される）。

#### 【0084】

操作部210からデバッグログステータス画面（図13）を開く操作が行われると、UIアプリケーション301は、デバッグログステータス画面を操作部210のタッチパネル付き表示部に表示する（S1701）。20

#### 【0085】

また、USBメモリがMFP100に装着されると、USBIF管理モジュール390は、USBメモリの接続を認識し、その旨をデバッグログ管理モジュール382に通知する（S1702）。なお、S1701とS1702の順序は逆でもよい。

#### 【0086】

また、UIアプリケーション301は、USBメモリ転送ボタン490が選択されたことを検知すると、USBメモリ転送ボタン490を反転表示する（S1703）。この状態で実行ボタン402が押下されたことを検知すると、UIアプリケーション301は、その旨をデバッグログ管理モジュール382に通知する（S1704）。30

#### 【0087】

上記S1702の通知及びS1704の通知を受けると、デバッグログ管理モジュール382は、ログ転送モジュール1050を起動する。

次に、S1705において、ログ転送モジュール1050は、デバッグログ用RAM領域1040の現時点のデバッグログをHDD208にファイル化する。

次に、S1706において、ログ転送モジュール1050は、上記S1705でHDD208にファイル化されたデバッグログ及びHDD208に自動退避されていたログファイルBも含めて、USBメモリ240に全てのログファイルを転送し、処理を終了する。

#### 【0088】

図16は、デバッグログの動作設定画面の一例を示す図である。

図13のデバッグログステータス画面に表示されるデバッグログ設定は、図16の動作設定画面から行われる。この動作設定画面も、図13に示したデバッグログステータス画面と同じく、MFP100の機能510別に設定が分かれており、520でログ自動退避（自動取得）のOff/On切り替え、530で自動退避のトリガ事象の設定が可能である。

#### 【0089】

自動取得520、及び、取得モード530の選択は、501及び502に示すプルダウンメニューから所望の値を選択し、実行ボタン502を押下することで可能である。上述のように自動取得及び取得モードが選択されて実行ボタン502を押下されると、選択さ4050

れた設定をデバッグログ管理モジュール382が読み込み、デバッグログ動作に反映する。このとき、デバッグログ管理モジュール382内のログ動作設定1010には、U/I画面から選択された条件に対応するログ設定ファイルが選択され、読み込まれる。

#### 【0090】

なお、ログ設定ファイル群A1には、MFPの各機能ごとに、JAM用のログ設定ファイル、ソフト例外用のログ設定ファイル、サービスコール用のログ設定ファイル、リブート用のログ設定ファイルがそれぞれ含まれている。即ち、ログ設定ファイル群A1には、MFPの各機能ごと及び取得モードごとのログ設定ファイルがそれぞれ含まれている。そして、取得モード530として「JAM」、「ソフト例外」、「サービスコール」又は「リブート」が選択された場合、ログ設定読み込みモジュール1020は、これに対応するログ設定ファイルを、ログ設定ファイル群A1から選択して読み込む。例えば、FAX機能の自動取得が有効にされ、取得モードとしてJAMが選択されていた場合、ログ設定読み込みモジュール1020は、FAX機能のJAMに対応するログ設定ファイルを、ログ設定ファイル群A1から選択して読み込む。10

#### 【0091】

なお、取得モード530として「カスタム」が選択された場合、ログ設定読み込みモジュール1020は、USBメモリ240からカスタム設定ファイルA2を読み込み、ログ動作設定1010に反映する。つまり取得モード530として「カスタム」を選択することで、柔軟なデバッグログ動作設定をMFP100に送り込むことができる。20

#### 【0092】

また、取得モード530として「無効」が選択された機能については、ログ設定読み込みモジュール1020は、設定ファイルの読み込みを行わない。20

設定クリアボタン550は、MFP100のデバッグログ動作設定を標準状態に戻すためのボタンである。この設定クリアボタン550が押下されると、デバッグログ管理モジュール382は、HDD等に格納されたデバッグログ動作設定の初期値等を読み出してログ動作設定1010に設定する。20

#### 【0093】

ログファイルクリアボタン551は、MFP100内(HDD208のログ永続化領域)に退避、保持されているデバッグログ関連のファイルやデータを全て削除する処理を実行するボタンである。30

#### 【0094】

これらの設定クリアボタン550、ログファイルクリアボタン551は、コントローラソフトウェア300の不具合調査が完了し、デバッグログ動作が不要となった時点でMFPを標準状態に戻すためのものである。

#### 【0095】

戻るボタン503は、図示しない1階層元(1つ前)のU/I画面に遷移するためのボタンである。

<デバッグログ動作のステータス表示>

前述のデバッグログの自動退避機能の実施に伴って、その動作状況をユーザ通知する表示が必要となる。そのU/I表示例を図17、図18で説明する。40

#### 【0096】

図17、図18は、通常のコピー機能画面の一例を示す図である。

本実施例のMFP100では、図17の610、図18の710に示す、操作画面下部の表示領域が、MFP全体の動作状況をユーザに通知するための領域として使用される。

図17の610では、ログのHDD退避中にMFP100の電源OFFをされるとHDD208の破損につながる可能性があるため、MFP100の電源OFFしなよう注意を促す表示がなされている例が示されている。

#### 【0097】

図18の710では、ログの自動退避が作動しHDD208にログファイルBが存在していることを通知する表示がなされている例が示されている。自動退避が作動したという50

ことは、即ちデバッグログを取得してコントローラソフトウェア300の動作を調査すべき状態が発生したということである。そのため、迅速にHDD208のログファイルBを回収し、コントローラソフトウェア300の開発者が分析することが望まれるため、ログ回収を促すための表示が操作画面下部の表示領域になされる（図18の710）。

#### 【0098】

なお、コピー画面内の各ボタンや機能の詳細は、本発明とは関係しないため説明は省略する。

なお、図17の610、図18の710に示す表示は、UIアプリケーション301が、デバッグログ管理モジュール382からの通知に応じて行うものである。

<設定からログ自動退避の流れ>

以下、図19を用いて、デバッグログ設定からログ自動退避までの処理の流れを説明する。

#### 【0099】

図19は、本実施例のデバッグログ自動退避処理のフローチャートである。このフローチャートの処理は、デバッグログ管理モジュール382により実行される（即ち、MainCPU205がBootROM207から読み出されたファームウェアを実行することにより実現される）。

#### 【0100】

デバッグログ管理モジュール382は、図16に示したデバッグログの動作設定画面からログの自動取得対象機能及び取得モードの設定が行われたことを検知すると（S1501）、S1502に処理を進める。

#### 【0101】

S1502では、デバッグログ管理モジュール382は、前記S1501でなされた設定に対応するログ設定ファイルを読み込んでログ動作設定1010に設定し、設定反映に必要な内部モジュールの再起動を行う。これにより、上述の前記S1501でなされた設定に対応するデバッグログ出力設定、自動退避トリガ設定がMFPに設定される。

#### 【0102】

そして、図3に示したMFP100の各モジュールは、前記S1501でなされた設定に対応する動作設定の元で動作を継続する（定常動作）（S1503）。

そして、ログ自動退避のトリガ事象が発生すると、エラーマネージャ385は、デバッグログ管理モジュール382に事象発生を通知する（S1504）。なお、この事象発生の通知を受けると、デバッグログ管理モジュール382は、ログ動作設定1010と発生した事象を比較する。そして、デバッグログ管理モジュール382が、発生した事象がログ退避トリガであると判断すると、ログ自動退避モジュール1030が、デバッグログ用RAM領域1040内の情報（ログ）を、一旦、HDD208の一時領域にファイル出力する。

#### 【0103】

次に、S1505において、ログパターンチェックモジュール1060が、HDD208の一時領域に出力したログ（ファイル）に、期待ログパターンが含まれているか否かを判定する。そして、HDD208の一時領域に出力したログ（ファイル）に、期待ログパターンが含まれていないと判断した場合（S1506でNo）、ログ自動退避モジュール1030は、上記一時領域に出力したログ（ファイル）を破棄する（S1515）。そして、図3に示したMFPの各モジュールは、再び定常動作に戻る（S1503）。

#### 【0104】

一方、HDD208の一時領域に出力したログ（ファイル）に、期待ログパターンが含まれていると判断した場合（S1506でYes）、ログ自動退避モジュール1030は、S1507処理を進める。

#### 【0105】

S1507では、ログ自動退避モジュール1030は、HDD208のログ永続化領域に空きがあるかを判定する。

10

20

30

40

50

そして、HDD208のログ永続化領域に空きがあると判定した場合(S1507でYes)、ログ自動退避モジュール1030は、S1508に処理を進める。

S1508では、ログ自動退避モジュール1030は、HDD208の一時領域に出力したログ(ファイル)をHDD208のログ永続化領域に退避し、S1509において、対応する機能の期待ログ保存フラグをOnにする。詳細には、上記S1506でログ(ファイル)内に含まれていると判定された期待ログパターンが設定されている機能の期待ログ保存フラグをOnにする。そして、図3に示したMFPの各モジュールは、再び定常動作に戻る(S1503)。

#### 【0106】

一方、HDD208のログ永続化領域に空きがないと判定した場合(S1507でNo)、ログ自動退避モジュール1030は、S1510に処理を進める。

S1510では、ログ自動退避モジュール1030は、ログ動作設定1010に保持されている退避ログの上書きモードが上書き設定On(上書き有効)か否かを判定する。そして、上書きモードが上書き設定On(上書き有効)と判定した場合(S1510でYes)、ログ自動退避モジュール1030は、S1513に処理を進める。

#### 【0107】

S1513では、ログ自動退避モジュール1030は、HDD208のログ永続化領域に存在するログファイルBのうち、最古のログファイルを、HDD208の一時領域に出力したログ(ファイル)で上書きする。さらに、S1514において、ログ自動退避モジュール1030は、対応する機能の期待ログ保存フラグをOnにする。詳細には、上記S1506でログ(ファイル)内に含まれていると判定された期待ログパターンが設定されている機能の期待ログ保存フラグをOnにする。そして、図3に示したMFPの各モジュールは、再び定常動作に戻る(S1503)。

#### 【0108】

なお、上記S1504で、上記事象発生の通知を受けると、デバッグログ管理モジュール382は、デバッグログ処理中となったことを、UIアプリケーション301に通知する。この通知を受けて、UIアプリケーション301は、操作部210に図17のMFPステータス表示を行う。そして、上記S1509又はS1514の処理を完了すると、デバッグログ管理モジュール382は、デバッグログ処理が終了したことを、UIアプリケーション301に通知する。この通知を受けて、UIアプリケーション301は、操作部210に図18のMFPステータス表示を行う。

#### 【0109】

一方、上記S1510において、上書きモードが上書き設定Off(上書き無効)と判定した場合(S1510でNo)、ログ自動退避モジュール1030は、S1511に処理を進める。

#### 【0110】

S1511では、ログ自動退避モジュール1030は、HDD208の一時領域に出力したログ(ファイル)を破棄し、S1512において、ログ動作設定1010内のログ自動退避設定をOffし、処理を終了する。なお、S1512でログ自動退避設定をOffするのは、退避領域がフルになっていて且つ上書きしない設定のため、この状態でデバッグログ動作をしても無駄な動作となってしまうため、このような無駄な動作を避けるためである。

#### 【0111】

以上でデバッグログ取得設定からログファイル自動退避までの処理が完結する。続いて、図15を用いて説明した、USBメモリでのログファイル回収を行うことで、所望のデバッグログを回収することができる。

#### 【0112】

##### <FTPサーバでの回収>

以下、USBメモリではなく、MFPとネットワーク接続されたサーバへ退避されたログを転送、回収する実施例について、図20及び図9の1830を用いて説明する。

10

20

30

40

50

**【0113】**

図20は、本実施例でFTPサーバにログ回収する処理のフローチャートである。なお、このフローチャートの処理は、UIアプリケーション301、USBIF管理モジュール390、デバッグログ管理モジュール382等により実行される（即ち、MainCP U205がBootROM207から読み出されたファームウェアを実行することにより実現される）。

**【0114】**

なお、デバッグログ管理モジュール382内のログ転送モジュール1050は、USBメモリの他にネットワーク接続された外部サーバへログファイルを転送する機能も備える。本実施例では、ログ転送モジュール1050が、FTPプロトコルを用いて、FTPサーバへ転送する構成を例に説明するが、他のプロトコルを用いて他の種類のサーバへ転送する構成であってもよい。また、サーバでなくとも、MFP100と通信可能な外部装置、例えば、NAS（Network Attached Storage）であってもよい。10

**【0115】**

操作部210からデバッグログステータス画面（図13）を開く操作が行われると、UIアプリケーション301は、デバッグログステータス画面を操作部210のタッチパネル付き表示部に表示する（S2001）。

**【0116】**

また、UIアプリケーション301は、FTPサーバ転送ボタン492が選択されたことを検知すると、FTPサーバ転送ボタン492を反転表示する（S2002）。この状態で実行ボタン402が押下されたことを検知すると、UIアプリケーション301は、その旨をデバッグログ管理モジュール382に通知する（S2003）。20

**【0117】**

上記S2003の通知を受けると、デバッグログ管理モジュール382は、ログ転送モジュール1050を起動する。

次に、S2004において、ログ転送モジュール1050は、ログ動作設定1010内のサーバ設定（具体的には、図9のサーバ設定1834）を用いてFTPサーバ250への接続確認を行う。

**【0118】**

そして、FTPサーバ250への接続に失敗した（接続NG）場合（S2004でNo）、ログファイルの転送は不可能なため、S2007において、ログ転送モジュール1050は、接続NGをステータスボタン493に表示し、処理を終了する。30

**【0119】**

一方、FTPサーバ250への接続に成功した（接続OK）場合（S2004でYes）、ログ転送モジュール1050は、S2005に処理を進める。

S2005では、ログ転送モジュール1050は、USBメモリ転送と同様に、デバッグログ用RAM領域1040の現時点のデバッグログをHDD208にファイル化する。

次に、S2006において、ログ転送モジュール1050は、上記S2005でHDD208にファイル化されたデバッグログ及びHDD208に自動退避されていたログファイルBも含めて、FTPサーバ250に全てのログファイルを転送し、処理を終了する。40

**【0120】**

<デバッグログ回収完了時>

デバッグログの取得が不要になった時点で、デバッグログ設定、及び、残ったログファイルのクリアなど、MFP100の動作を標準状態に戻す必要がある。

**【0121】**

そのために図16で説明した設定クリアボタン550、ログファイルクリアボタン551が用意されており、これらのボタンを押下してクリア実行する。この操作によって、MFP100はデバッグログ取得設定する前の、標準動作状態に戻るため、以後、再度、明示的にデバッグログ動作設定を行わない限り、コントローラソフトウェア300のデバッグログ管理モジュール382は動作することはない。50

**【 0 1 2 2 】**

また、上記実施例では、本発明の情報処理装置を、一実施例として、MFP等の画像処理装置を用いて説明したが、本発明の情報処理装置は画像処理装置に限られるものではない。ファームウェアにより動作する情報処理装置であれば、どのような情報処理装置であつてもよい。

**【 0 1 2 3 】**

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。10

**【 0 1 2 4 】**

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。

**(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等のプロセッサ）がプログラムを読み出して実行する処理である。

**【 0 1 2 5 】**

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。20

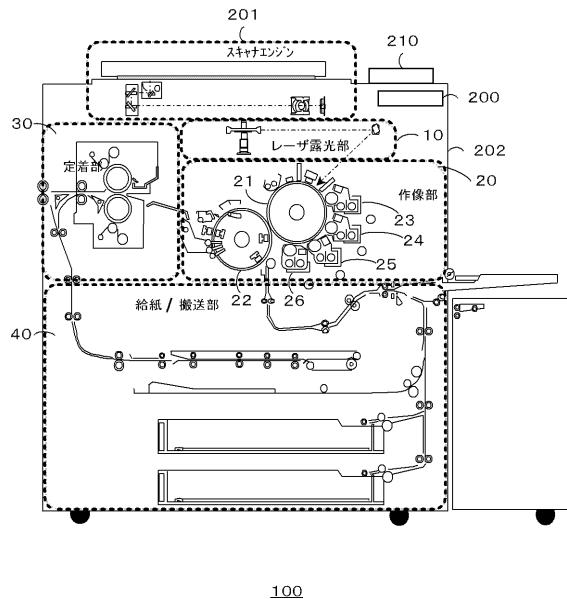
**【 0 1 2 6 】**

以上説明したように、本発明によれば、デバッグログ取得（ファイル化）の自動化と、非開発者にも理解可能な取得設定および取得ステータスUIを備えることで、情報処理装置の制御ファームウェアにおける所望のデバッグログの取得の確率を向上し、ファームウェアの不具合調査の効率を高めることができる。30

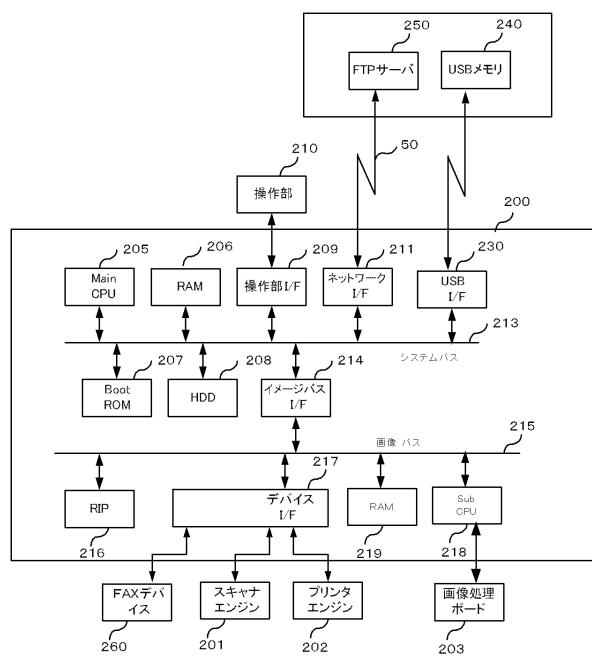
**【 符号の説明】****【 0 1 2 7 】**

1 0 0	MFP
2 0 0	コントローラユニット
2 1 0	操作部
2 0 1	スキャナエンジン
2 0 2	プリンタエンジン
2 0 3	画像処理ボード
2 6 0	FAXデバイス
3 0 0	コントローラソフトウェア
3 8 2	デバッグログ管理モジュール
3 8 5	エラーマネージャ

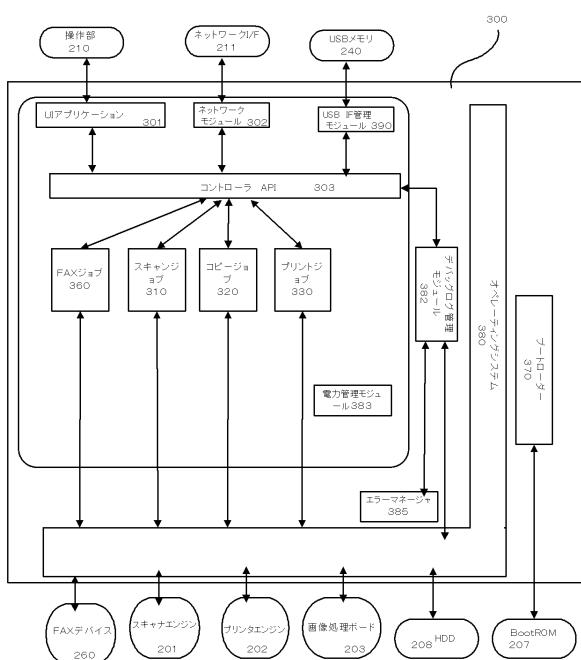
【図1】



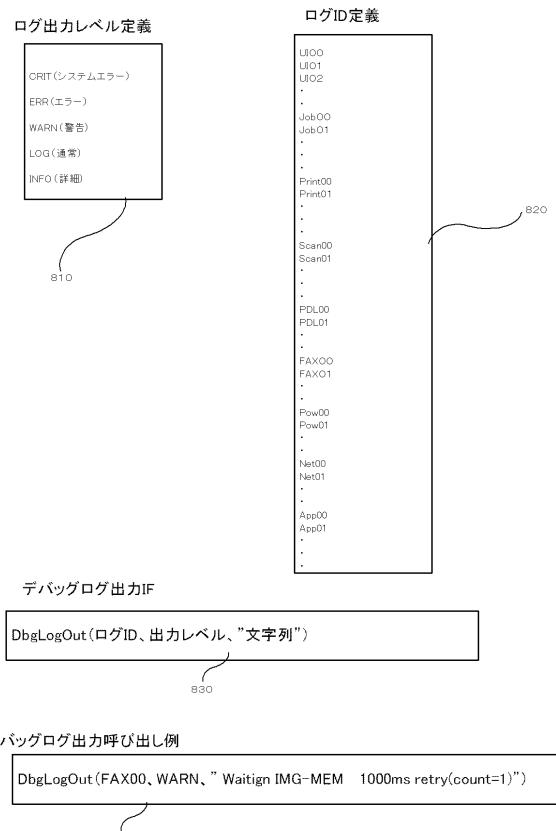
【図2】



【図3】



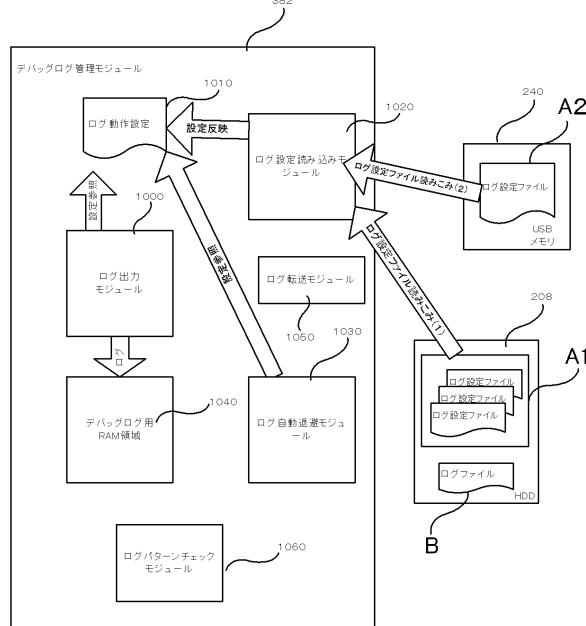
【図4】



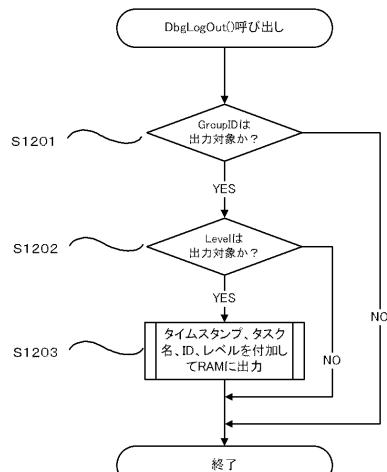
【図5】

line	time	TASK	GroupID	LEVEL	Message
1	315 85308	NetTASK	Net13	WARN	unknown packet
2	315 85309	NetTASK	Net13	WARN	packet ignore!
3	316 85308	NetTASK	Net13	WARN	unknown packet
4	316 85309	NetTASK	Net13	WARN	packet ignore!
5	317 85308	NetTASK	Net13	WARN	unknown packet
6	317 85309	NetTASK	Net13	WARN	packet ignore!
7	318 042	NetTASK	Net11	WARN	unknown packet_rejected
8	318 82693	UIJob	UI02	LOG	[UI] StartKey_Pressed!
9	318 82694	UIJob	UI02	LOG	CopyJob_Kind_JobMode=Normal ColorMode=full
10	318 85218	Cpy_App	App03	LOG	CopyJob API call start(mode=normal)
11	318 85202	Cpy_App	App03	LOG	JobAttr1 = 31
12	318 85313	Cpy_App	App03	LOG	JobAttr2 = 10
13	318 85200	NetTASK	Net13	WARN	unknown packet
14	318 85209	NetTASK	Net13	WARN	packet ignore!
15	318 85202	Cpy_App	App03	LOG	JobAttr2 = 1683
16	318 85202	Cpy_App	App03	LOG	JobAttr4 = 0
17	319 02509	Fax_App	App03	LOG	FAX.comm received establishconnection.
18	319 02537	FAXAapp	App03	LOG	[FAX] request FAX-Comm-JOB
19	320 10785	TskJob	JOB00	LOG	CopyJob_Create/JobID=0x10005001
20	320 12231	TskJob	Job10	LOG	###[#]CopyJobJob.StateChange State(1)
21	320 12093	UIJob	UI10	LOG	[UI] Update_Status_Copy(1)
22	320 15093	TskJob	JOB00	LOG	create ScanSubJob_start(0x10005001)
23	320 15094	TskJob	JOB00	LOG	create PrintSubJob_start(0x10005001)
24	320 16434	TskIMG	IMG01	LOG	Initializing IMG-BUFFER ADDR=0x50004000
25	320 18233	TskScn	Scn00	LOG	ScannerConnect OK_Start Position Ready
26	320 18200	TskScn	Scn00	LOG	ScannerLamp OK_Start IMG read(mode=0x81)
27	320 18231	TskJob	Job10	LOG	###[#]CopyJobJob.StateChange State(2)
28	320 18232	TskScn	Scn00	INFO	Param[0]_0x23 a0 32
29	320 18233	TskScn	Scn00	INFO	Param[1]_0x23 a0 32
30	320 18234	FaxDev	FAX00	WARN	Waitimg IMG-MEM 1000ms retry(count=1)
31	320 18234	UIJob	UI10	LOG	[UI] Update_Status_Copy(2)
32	320 18234	TskScn	Scn00	INFO	Param[2]_80 00 ff
33	320 18234	TskScn	Scn00	INFO	Param[3]_82 8b 83 ff
34	320 18236	PowTask	Pow00	ERR	illegal command ignore_OPE(PowerOFF)
35	320 18237	TskScn	Scn00	INFO	Param[4]_ff ff ff ff
36	320 18509	TskIMG	IMG02	LOG	Get IMG from Scanner ADDR=0x50004000
37	320 18513	TskIMG	IMG02	LOG	IMG header 0x80000b01
38	320 22025	FaxDev	FAX00	WARN	Waitimg IMG-MEM 1000ms retry(count=2)
39	321 22023	FaxDev	FAX00	WARN	Waitimg IMG-MEM 1000ms retry(count=3)
40	321 85308	NetTASK	Net13	WARN	unknown packet
41	321 85309	NetTASK	Net13	WARN	packet ignore!
42	322 22022	FaxDev	FAX00	WARN	Waitimg IMG-MEM 1000ms retry(count=4)
43	322 85308	NetTASK	Net13	WARN	unknown packet
44	322 85309	NetTASK	Net13	WARN	packet ignore!
45	322 22023	FaxDev	FAX00	WARN	Waitimg IMG-MEM 1000ms retry(count=5)
46	323 25233	TskScn	Scn00	LOG	ScannerRead_Finish
47	323 28509	TskIMG	IMG02	LOG	IMG Stored ADDR=0x50004000
48	323 85308	NetTASK	Net13	WARN	unknown packet
49	323 85309	NetTASK	Net13	WARN	packet ignore!
50	323 96233	TskPnm	Pnm00	LOG	PrinterCheck_Start
51	323 96231	TskJob	Job10	LOG	###[#]CopyJobJob.StateChange State(3)
52	323 96232	TskPnm	Pnm00	LOG	CST[1] NO PAPER
53	323 96243	TskJob	Job10	LOG	CST[2] NO PAPER
54	323 96243	TskPnm	Pnm00	LOG	[CopyJob]CST Changed CST(3)
55	323 96244	TskPnm	Pnm03	WARN	CST[4] NO PAPER
56	323 96252	TskPnm	Pnm10	LOG	CST CHANGE CST[1] to CST[3]
57	323 96231	TskJob	Job10	LOG	[CopyJob]CST Changed CST(3)
58	324 028	TskPnm	Pnm20	LOG	IMG ADDR=0x50004000
59	324 22023	FaxDev	FAX00	WARN	Waitimg IMG-MEM 1000ms retry(count=6)
60	324 28246	TskPnm	Pnm03	LOG	Paper Feed Start!

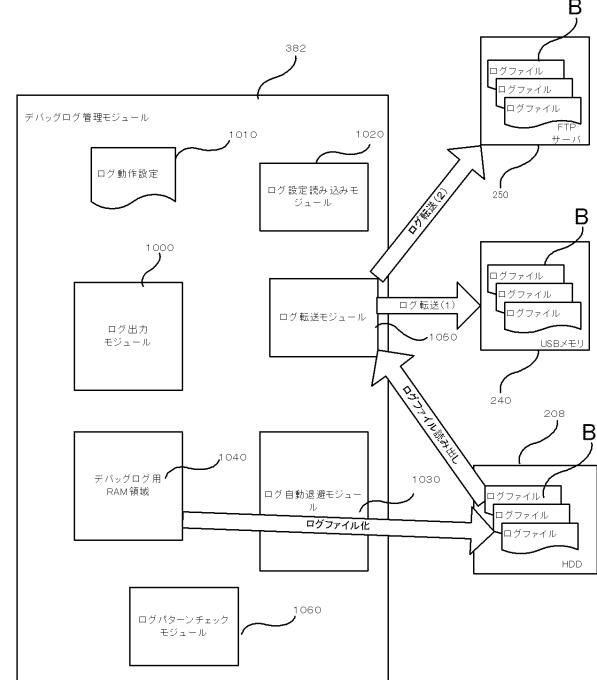
【図6】



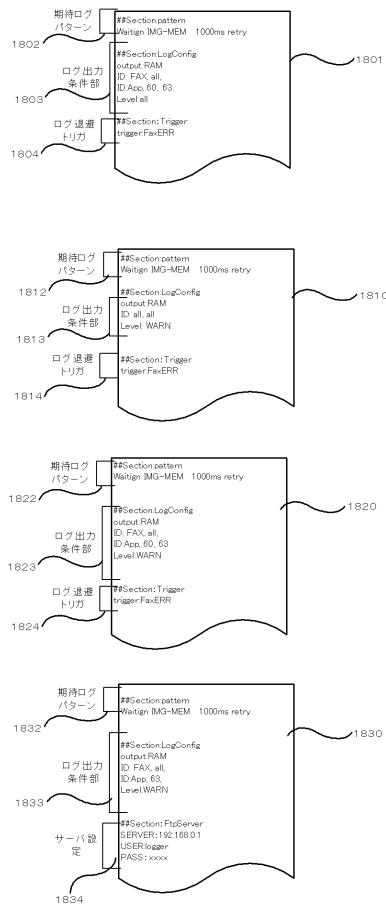
【図7】



【図8】



【図9】



【図10】

line	time	TASK	GroupID	LEVEL	Message
1	319.02509	FaxApp	App60	LOG	FAX comm received establishconnection.
2	319.02537	FaxApp	App63	LOG	[FAX] request FAX-Comm-JOB
3	320.15235	FaxDev	FAX00	WARN	Waiting IMG-MEM retry(count=1)
4	320.15265	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=1)
5	321.22023	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=2)
6	322.22023	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=3)
7	323.22023	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=4)
8	324.22023	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=5)
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					
24					
25					
26					
27					
28					
29					
30					
31					
32					
33					
34					
35					
36					
37					
38					
39					
40					
41					
42					
43					
44					
45					
46					

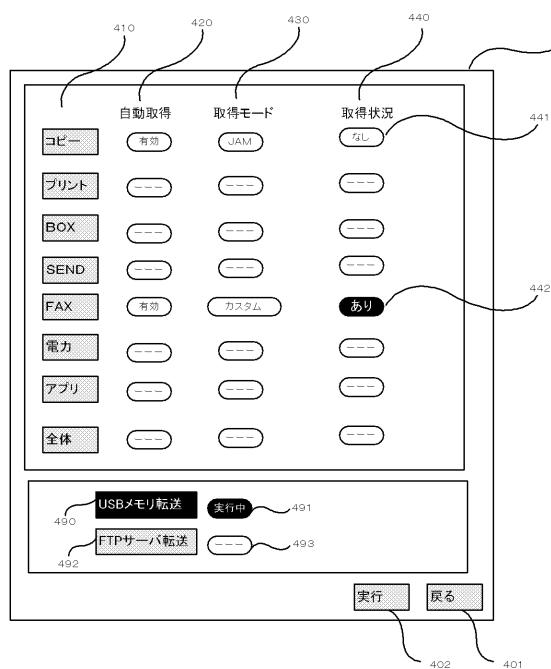
【図11】

line	time	TASK	GroupID	LEVEL	Message
1	315.85308	NetTASK	Net13	WARN	unknown packet
2	316.85309	NetTASK	Net13	WARN	packet ignore!
3	316.85308	NetTASK	Net13	WARN	unknown packet
4	316.85309	NetTASK	Net13	WARN	packet ignore!
5	317.85308	NetTASK	Net13	WARN	unknown packet
6	317.85309	NetTASK	Net13	WARN	packet ignore!
7	318.042	NetTASK	Net11	WARN	unknown packet rejected
8	318.85308	NetTASK	Net13	WARN	unknown packet
9	318.85309	NetTASK	Net13	WARN	packet ignore!
10	320.18234	FaxDev	FAX00	WARN	Waiting IMG-MEM retry(count=1)
11	320.18236	PowTsl	Power0	ERR	illegal command ignore OPE[Power OFF]
12	320.22025	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=1)
13	321.22023	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=2)
14	321.85308	NetTASK	Net13	WARN	unknown packet
15	321.85308	NetTASK	Net13	WARN	packet ignore!
16	322.22022	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=3)
17	322.85308	NetTASK	Net13	WARN	unknown packet
18	322.85309	NetTASK	Net13	WARN	packet ignore!
19	323.22023	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=4)
20	323.85308	NetTASK	Net13	WARN	unknown packet
21	323.85309	NetTASK	Net13	WARN	packet ignore!
22	323.8524	TslPrn	Prn03	WARN	CST[1] NO PAPER
23	323.85243	TslPrn	Prn03	WARN	CST[2] NO PAPER
24	323.85244	TslPrn	Prn03	WARN	CST[4] NO PAPER
25	324.22023	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=5)
26					
27					
28					
29					
30					
31					
32					
33					
34					
35					
36					
37					
38					
39					
40					
41					
42					
43					
44					
45					
46					

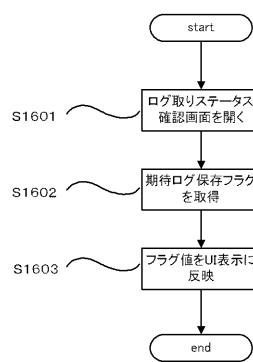
【図12】

line	time	TASK	GroupID	LEVEL	Message
1	320.18234	FaxDev	FAX00	WARN	Waiting IMG-MEM retry(count=1)
2	320.22025	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=1)
3	321.22023	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=2)
4	322.22022	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=3)
5	323.22023	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=4)
6	324.22023	FaxDev	FAX00	WARN	Waiting IMG-MEM 1000ms retry(count=5)
7					
8					
9					
10					
11					
12					
13					
14					
15					
16					
17					
18					
19					
20					
21					
22					
23					
24					
25					
26					
27					
28					
29					
30					
31					
32					
33					
34					
35					
36					
37					
38					
39					
40					
41					
42					
43					
44					
45					
46					

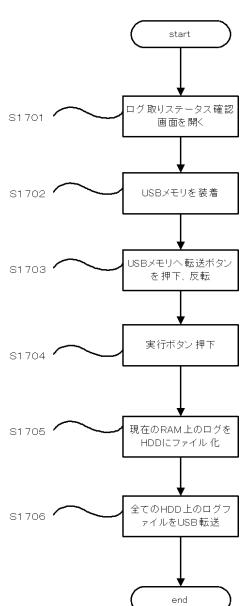
【図13】



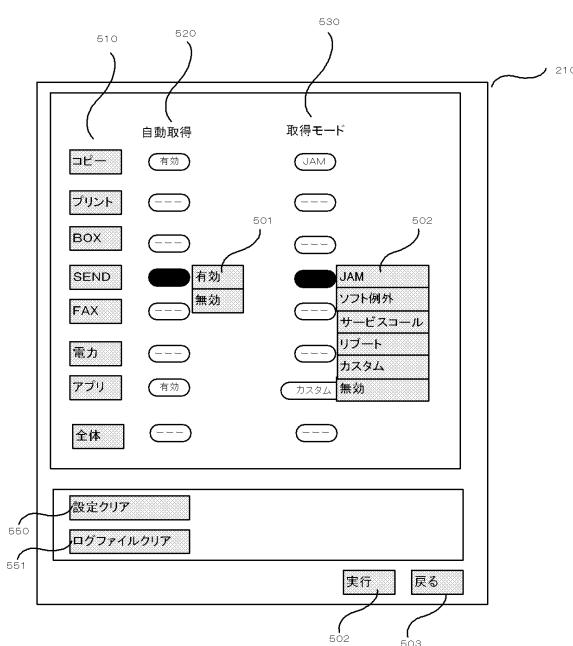
【図14】



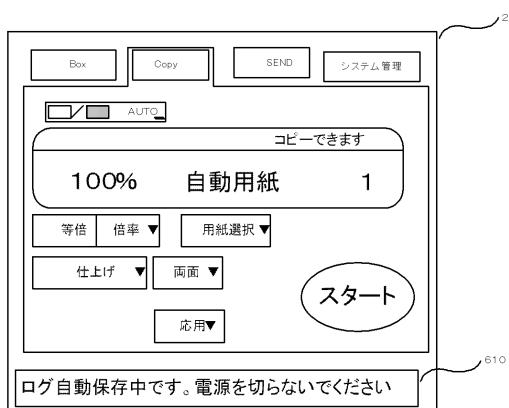
【図15】



【図16】



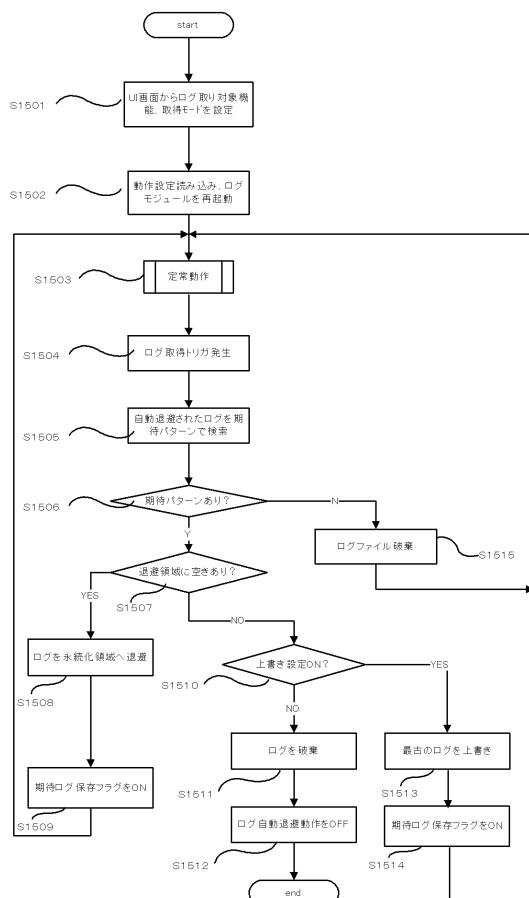
【図17】



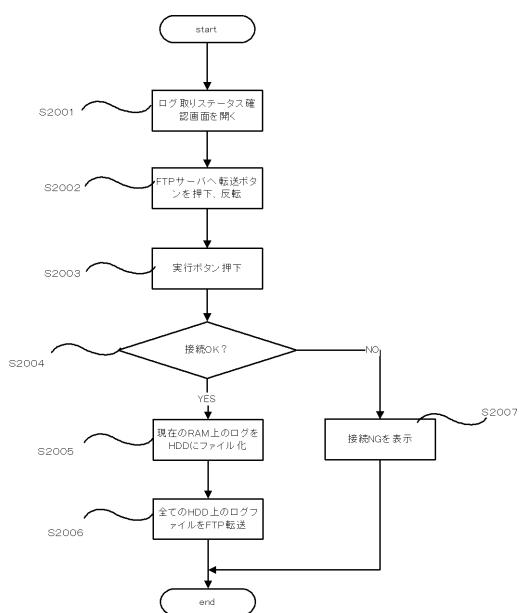
【図18】



【図19】



【図20】



---

フロントページの続き

(56)参考文献 特開2010-056744(JP,A)  
特開2007-233918(JP,A)  
特開2001-257823(JP,A)

(58)調査した分野(Int.Cl., DB名)

H 04 N 1 / 00