

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4128516号
(P4128516)

(45) 発行日 平成20年7月30日(2008.7.30)

(24) 登録日 平成20年5月23日(2008.5.23)

(51) Int.Cl. F I
H04N 1/00 (2006.01) H04N 1/00 C

請求項の数 30 (全 34 頁)

(21) 出願番号	特願2003-364032 (P2003-364032)	(73) 特許権者	000006747
(22) 出願日	平成15年10月24日(2003.10.24)		株式会社リコー
(65) 公開番号	特開2004-194298 (P2004-194298A)		東京都大田区中馬込1丁目3番6号
(43) 公開日	平成16年7月8日(2004.7.8)	(74) 代理人	100070150
審査請求日	平成17年11月24日(2005.11.24)		弁理士 伊東 忠彦
(31) 優先権主張番号	特願2002-333966 (P2002-333966)	(72) 発明者	杉下 悟
(32) 優先日	平成14年11月18日(2002.11.18)		東京都大田区中馬込1丁目3番6号 株式
(33) 優先権主張国	日本国(JP)		会社リコー内
(31) 優先権主張番号	特願2002-342827 (P2002-342827)	(72) 発明者	阿部 良彦
(32) 優先日	平成14年11月26日(2002.11.26)		東京都大田区中馬込1丁目3番6号 株式
(33) 優先権主張国	日本国(JP)		会社リコー内
(31) 優先権主張番号	特願2002-348317 (P2002-348317)	(72) 発明者	加藤 良一
(32) 優先日	平成14年11月29日(2002.11.29)		東京都大田区中馬込1丁目3番6号 株式
(33) 優先権主張国	日本国(JP)		会社リコー内

最終頁に続く

(54) 【発明の名称】 画像形成装置、プログラム更新方法

(57) 【特許請求の範囲】

【請求項1】

画像形成に係る処理を行う1つ以上の処理部と、前記処理部と直接または他の処理部を介して通信する制御部と、前記処理部と前記制御部とを動作させるプログラムとを有する画像形成装置において、

前記プログラムを更新する更新プログラムを取得する更新プログラム取得部と、

一の処理部の前記プログラムを、該一の処理部と前記制御部とが通信をするために経由する処理部の数に応じて定まる順番で、前記更新プログラムに更新させるプログラム更新部と

を有することを特徴とする画像形成装置。

10

【請求項2】

前記プログラム更新部は、前記更新プログラム取得部が取得した更新プログラムに基づき、前記順番を定める更新順番情報を生成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】

前記更新順番情報は、前記順番と、取得した更新プログラムにより更新されるプログラムの種類とを対応付けた情報を有することを特徴とする請求項2に記載の画像形成装置。

【請求項4】

前記更新順番情報は、取得した更新プログラムにより更新されるプログラムの更新状況を示す情報を有することを特徴とする請求項2に記載の画像形成装置。

20

【請求項 5】

前記制御部は、画像形成処理に係る処理を実行する1つ以上のアプリケーションを有し、前記プログラム更新部は、各アプリケーションのプログラムを個々に更新することが可能であ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 6】

前記プログラム更新部は、前記画像形成装置を操作するオペレータに対して該画像形成装置に関する情報を表示する処理を行う処理部の前記プログラムを更新する順番を最後の順番と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 7】

ネットワークと接続するための通信部をさらに有し、

前記更新プログラム取得部は、前記通信部により前記更新プログラムを取得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 8】

前記制御部は、前記プログラムの更新が終了した処理部の順に該処理部を再起動させることを特徴とする請求項7に記載の画像形成装置。

【請求項 9】

前記制御部は、前記プログラムを更新する全ての処理部が更新を終了した後に、前記プログラムを更新した処理部を再起動させることを特徴とする請求項7に記載の画像形成装置。

【請求項 10】

前記制御部は、自らのプログラムを更新する際に、該プログラムが格納された記憶領域とは異なる他の記憶領域に格納された更新用プログラムで動作することを特徴とする請求項7に記載の画像形成装置。

【請求項 11】

前記更新用プログラムは、揮発性記憶部の領域に格納されることを特徴とする請求項10に記載の画像形成装置。

【請求項 12】

前記更新用プログラムのサイズは、前記制御部のプログラムのサイズより小さいことを特徴とする請求項11に記載の画像形成装置。

【請求項 13】

前記制御部は、前記処理部のプログラムを更新する際に、該処理部に対し、処理の実行を制限させることを特徴とする請求項7に記載の画像形成装置。

【請求項 14】

前記制御部は、前記プログラムを更新した処理部との通信が切断すると、自らを再起動することを特徴とする請求項7に記載の画像形成装置。

【請求項 15】

前記制御部は、再起動をさせる前記処理部のうち、最後に処理部を再起動させてから所定の時間が経過した後に自らを再起動することを特徴とする請求項7に記載の画像形成装置。

【請求項 16】

前記制御部は、自らが再起動する際に、画像形成装置に係る装置状態を不揮発性記憶部に記憶し、前記制御部の再起動後、記憶した前記装置状態に前記画像形成装置の状態を遷移させることを特徴とする請求項7に記載の画像形成装置。

【請求項 17】

前記装置状態は、前記画像形成装置の消費電力を低減する状態である省電力状態であることを特徴とする請求項16に記載の画像形成装置。

【請求項 18】

前記画像形成装置に関する情報を表示する表示部を有し、

前記装置状態は、前記表示部に表示されている情報の状態であることを特徴とする請求項16に記載の画像形成装置。

10

20

30

40

50

【請求項 19】

前記表示部は、前記プログラムの更新の進捗状況を表示することを特徴とする請求項 18 に記載の画像形成装置。

【請求項 20】

前記表示部は、前記画像形成装置の消費電力を低減する状態であれば、前記プログラムの更新の進捗状況を表示しないことを特徴とする請求項 19 に記載の画像形成装置。

【請求項 21】

前記プログラム更新部は、取得した更新プログラムに関する情報を含む更新管理テーブルを生成することを特徴とする請求項 7 に記載の画像形成装置。

【請求項 22】

前記更新管理テーブルは、取得した更新プログラムを特定するモジュール ID を含むことを特徴とする請求項 21 に記載の画像形成装置。

【請求項 23】

前記処理部は、前記制御部が直接に再起動可能な処理部と、前記制御部が前記処理部に対して再起動要求を行うことにより再起動可能な処理部との 2 つの種類があることを特徴とする請求項 7 に記載の画像形成装置。

【請求項 24】

前記プログラムを更新した結果を示す更新結果情報を生成する更新結果情報生成部と、前記更新結果情報に基づいた更新結果を出力する更新結果出力部とをさらに有することを特徴とする請求項 7 に記載の画像形成装置。

【請求項 25】

前記更新結果出力部は、前記更新結果を印刷するか、電子メールで送信するか、画像形成装置に関する情報を表示する表示部に表示することを特徴とする請求項 24 に記載の画像形成装置。

【請求項 26】

前記更新結果情報は、更新前のプログラムのバージョンと、更新後のプログラムのバージョンと、更新した日時と、更新の成否のうち、いずれか 1 つ以上を含むことを特徴とする請求項 24 に記載の画像形成装置。

【請求項 27】

前記更新結果情報生成部は、前記更新結果情報を不揮発性記憶部に記憶することを特徴とする請求項 24 に記載の画像形成装置。

【請求項 28】

前記更新結果情報生成部は、過去に行った更新での更新結果情報に基づく更新履歴情報に、生成した更新結果情報を追記することを特徴とする請求項 27 に記載の画像形成装置。

【請求項 29】

前記制御部は、前記更新結果情報生成部が前記更新結果情報を不揮発性記憶部に記憶した後に、前記画像形成装置を再起動し、

前記更新結果出力部は、再起動後に前記更新結果を出力することを特徴とする請求項 27 に記載の画像形成装置。

【請求項 30】

画像形成に係る処理を行う 1 つ以上の処理部と、前記処理部と直接または他の処理部を介して通信する制御部と、前記処理部と前記制御部とを動作させるプログラムとを有する画像形成装置のプログラム更新方法であって、

前記プログラムを更新する更新プログラムを取得する更新プログラム取得段階と、

一の処理部の前記プログラムを、該一の処理部と前記制御部との通信をするために経由する処理部の数に応じて定まる順番で、前記更新プログラムに更新させるプログラム更新段階と

を有することを特徴とするプログラム更新方法。

【発明の詳細な説明】

10

20

30

40

50

【技術分野】

【0001】

本発明は、1つ以上のプログラムの更新に関し、特にプログラムを更新する画像形成装置、及びそのプログラム更新方法に関する。

【背景技術】

【0002】

近年、ファクシミリ、プリンタ、コピーおよびスキャナなどの各装置の機能を1つの筐体内に収納した画像形成装置（以下、融合機という）が知られるようになった。この融合機は、1つの筐体内に表示部、印刷部および撮像部などを設けると共に、ファクシミリ、プリンタ、コピーおよびスキャナにそれぞれ対応する4種類のアプリケーションを設け、そのアプリケーションを切り替えることより、ファクシミリ、プリンタ、コピーおよびスキャナとして動作させるものである。

10

【0003】

このように、融合機は、プログラムに基づき処理を実行する種々のアプリケーションやモジュールにより動作しているため、NV-RAMなどに書き込まれたプログラムを更新することにより、新たな機能の追加などを行うことができる。

【0004】

このプログラムの更新処理は、インターネットなどのネットワークを介し、更新プログラムが格納されたサーバに接続し更新プログラムを取得（以下、ダウンロードと記す）することにより行われることがある。

20

【発明の開示】

【発明が解決しようとする課題】

【0005】

このダウンロードした更新プログラムは、プリンタのプログラムを更新するものや、印刷用紙に関する制御を行うプログラムを更新するものなど、複数のプログラムが含まれ、それらが連続し、1つのデータのようにになっている。

【0006】

この更新プログラムのデータ構造は、例えば図37に示されるように、更新プログラム全体のヘッダである更新プログラムヘッダが先頭にあり、次に個々の更新プログラムのヘッダである個別更新プログラムヘッダと更新プログラム本体が、更新プログラムの数だけ連続するデータ構造となっている。

30

【0007】

更新プログラムヘッダの全長は、更新プログラムヘッダを除く全体のデータ長を表し、全プログラム数は、この更新プログラムに含まれる全ての更新プログラムの数を示す。また、個別更新プログラムヘッダのプログラム種類は、例えばプリンタ用プログラムなどのプログラムの種類を表し、バージョンは、更新プログラムのバージョンを表し、長は更新プログラム本体のデータ長を表す。

【0008】

このように連続した更新プログラムを、図37に示したデータ構造の順序通りにプログラムの更新を行うと、プログラムの更新ができなくなる可能性がある。

40

【0009】

また、サービスマンなどが画像形成装置を直接操作せず、ネットワークを介して行われるプログラムの更新の場合、プログラムを更新した画像形成装置の画面には、プログラムの更新が完了したことが表示されることがある。

【0010】

画像形成装置にそのような画面が表示されていると、オペレータが使用する際に戸惑う原因となる。また、プログラムの更新によりオペレータが電源を入れ直したりしなければならない。

【0011】

このように、サービスマンなどが画像形成装置を直接操作せず、ネットワークを介して

50

行われるプログラムの更新の場合、プログラムの更新が夜間などに自動的に行われることも相俟って、オペレータは、プログラムが更新されたことや更新結果を知ることができないか困難であった。

【0012】

本発明は、このような問題点に鑑み、プログラムをネットワークを介して更新する画像形成装置に好適なプログラム更新方法、そのプログラム更新方法を搭載した画像形成装置を提供することを目的とする。

【課題を解決するための手段】

【0013】

上記課題を解決するために、本発明は、画像形成に係る処理を行う1つ以上の処理部と、前記処理部と直接または他の処理部を介して通信する制御部と、前記処理部と前記制御部とを動作させるプログラムとを有する画像形成装置において、前記プログラムを更新する更新プログラムを取得する更新プログラム取得部と、一の処理部の前記プログラムを、該一の処理部と前記制御部とが通信をするために経由する処理部の数に応じて定まる順番で、前記更新プログラムに更新させるプログラム更新部とを有することを特徴とする。

10

【0014】

また、上記課題を解決するために、本発明は、前記プログラム更新部は、前記更新プログラム取得部が取得した更新プログラムに基づき、前記順番を定める更新順番情報を生成することを特徴とする。

【0015】

20

また、上記課題を解決するために、本発明は、前記更新順番情報は、前記順番と、取得した更新プログラムにより更新されるプログラムの種類とを対応付けた情報を有することを特徴とする。

【0016】

また、上記課題を解決するために、本発明は、前記更新順番情報は、取得した更新プログラムにより更新されるプログラムの更新状況を示す情報を有することを特徴とする。

【0017】

また、上記課題を解決するために、本発明は、前記制御部は、画像形成処理に係る処理を実行する1つ以上のアプリケーションを有し、前記プログラム更新部は、各アプリケーションのプログラムを個々に更新することが可能であることを特徴とする。

30

【0018】

また、上記課題を解決するために、本発明は、前記プログラム更新部は、前記画像形成装置を操作するオペレータに対して該画像形成装置に関する情報を表示する処理を行う処理部の前記プログラムを更新する順番を最後の順番とすることを特徴とする。

【0019】

また、上記課題を解決するために、本発明は、ネットワークと接続するための通信部をさらに有し、前記更新プログラム取得部は、前記通信部により前記更新プログラムを取得することを特徴とする。

【0020】

また、上記課題を解決するために、本発明は、前記制御部は、前記プログラムの更新が終了した処理部の順に該処理部を再起動させることを特徴とする。

40

【0021】

また、上記課題を解決するために、本発明は、前記制御部は、前記プログラムを更新する全ての処理部が更新を終了した後に、前記プログラムを更新した処理部を再起動させることを特徴とする。

【0022】

また、上記課題を解決するために、本発明は、前記制御部は、自らのプログラムを更新する際に、該プログラムが格納された記憶領域とは異なる他の記憶領域に格納された更新プログラムで動作することを特徴とする。

【0023】

50

また、上記課題を解決するために、本発明は、前記更新用プログラムは、揮発性記憶部の領域に格納されることを特徴とする。

【0024】

また、上記課題を解決するために、本発明は、前記更新用プログラムのサイズは、前記制御部のプログラムのサイズより小さいことを特徴とする。

【0025】

また、上記課題を解決するために、本発明は、前記制御部は、前記処理部のプログラムを更新する際に、該処理部に対し、処理の実行を制限させることを特徴とする。

【0026】

また、上記課題を解決するために、本発明は、前記制御部は、前記プログラムを更新した処理部との通信が切断すると、自らを再起動することを特徴とする。

10

【0027】

また、上記課題を解決するために、本発明は、前記制御部は、再起動をさせる前記処理部のうち、最後に処理部を再起動させてから所定の時間が経過した後に自らを再起動することを特徴とする。

【0028】

また、上記課題を解決するために、本発明は、前記制御部は、自らが再起動する際に、画像形成装置に係る装置状態を不揮発性記憶部に記憶し、前記制御部の再起動後、記憶した前記装置状態に前記画像形成装置の状態を遷移させることを特徴とする。

【0029】

20

また、上記課題を解決するために、本発明は、前記装置状態は、前記画像形成装置の消費電力を低減する状態である省電力状態であることを特徴とする。

【0030】

また、上記課題を解決するために、本発明は、前記画像形成装置に関する情報を表示する表示部を有し、前記装置状態は、前記表示部に表示されている情報の状態であることを特徴とする。

【0031】

また、上記課題を解決するために、本発明は、前記表示部は、前記プログラムの更新の進捗状況を表示することを特徴とする。

【0032】

30

また、上記課題を解決するために、本発明は、前記表示部は、前記画像形成装置の消費電力を低減する状態であれば、前記プログラムの更新の進捗状況を表示しないことを特徴とする。

【0033】

また、上記課題を解決するために、本発明は、前記プログラム更新部は、取得した更新プログラムに関する情報を含む更新管理テーブルを生成することを特徴とする。

【0034】

また、上記課題を解決するために、本発明は、前記更新管理テーブルは、取得した更新プログラムを特定するモジュールIDを含むことを特徴とする。

【0035】

40

また、上記課題を解決するために、本発明は、前記処理部は、前記制御部が直接に再起動可能な処理部と、前記制御部が前記処理部に対して再起動要求を行うことにより再起動可能な処理部との2つの種類があることを特徴とする。

【0036】

また、上記課題を解決するために、本発明は、前記プログラムを更新した結果を示す更新結果情報を生成する更新結果情報生成部と、前記更新結果情報に基づいた更新結果を出力する更新結果出力部とをさらに有することを特徴とする。

【0037】

また、上記課題を解決するために、本発明は、前記更新結果出力部は、前記更新結果を印刷するか、電子メールで送信するか、画像形成装置に関する情報を表示する表示部に表

50

示することを特徴とする。

【0038】

また、上記課題を解決するために、本発明は、前記更新結果情報は、更新前のプログラムのバージョンと、更新後のプログラムのバージョンと、更新した日時と、更新の成否のうち、いずれか1つ以上を含むことを特徴とする。

【0039】

また、上記課題を解決するために、本発明は、前記更新結果情報生成部は、前記更新結果情報を不揮発性記憶部に記憶することを特徴とする。

【0040】

また、上記課題を解決するために、本発明は、前記更新結果情報生成部は、過去に行った更新での更新結果情報に基づく更新履歴情報に、生成した更新結果情報を追記することを特徴とする。

10

【0041】

また、上記課題を解決するために、本発明は、前記制御部は、前記更新結果情報生成部が前記更新結果情報を不揮発性記憶部に記憶した後に、前記画像形成装置を再起動し、前記更新結果出力部は、再起動後に前記更新結果を出力することを特徴とする。

【0042】

また、上記課題を解決するために、本発明は、画像形成に係る処理を行う1つ以上の処理部と、前記処理部と直接または他の処理部を介して通信する制御部と、前記処理部と前記制御部とを動作させるプログラムとを有する画像形成装置のプログラム更新方法であって、前記プログラムを更新する更新プログラムを取得する更新プログラム取得段階と、一の処理部の前記プログラムを、該一の処理部と前記制御部との通信をするために経由する処理部の数に応じて定まる順番で、前記更新プログラムに更新させるプログラム更新段階とを有することを特徴とする。

20

【0043】

また、上記課題を解決するために、本発明は、画像形成処理で使用されるハードウェア資源と、画像形成に係る処理を行う1つ以上の処理部と、該処理部を動作させるプログラムと、通信部とを有する画像形成装置のプログラム更新方法であって、前記プログラムを更新する更新プログラムを、前記通信部により取得する更新プログラム取得段階と、前記プログラムを、前記更新プログラムで更新するプログラム更新段階と、前記プログラムを更新した処理部を再起動させる再起動段階とを有することを特徴とする。

30

【0044】

また、上記課題を解決するために、本発明は、前記再起動段階は、前記プログラムの更新をした処理部の順に再起動することを特徴とする。

【0045】

また、上記課題を解決するために、本発明は、画像形成処理で使用されるハードウェア資源と、画像形成に係る処理を行うプログラムと、通信部とを有する画像形成装置のプログラム更新方法において、前記プログラムを更新する更新プログラムを前記通信部により取得する更新プログラム取得段階と、前記プログラムを、前記更新プログラムで更新するプログラム更新段階と、前記プログラムを更新した結果を示す更新結果情報を生成する更新結果情報生成段階と、前記更新結果情報に基づく更新結果を出力する更新結果出力段階とを有することを特徴とする。

40

【0046】

また、上記課題を解決するために、本発明は、前記更新結果出力段階では、前記更新結果を印刷するか、電子メールで送信するか、前記画像形成装置が有する表示部に表示することを特徴とする。

【0047】

また、上記課題を解決するために、本発明は、前記更新結果情報は、更新前のプログラムのバージョンと、更新後のプログラムのバージョンと、更新した日時と、更新の成否のうち、いずれか1つ以上を含むことを特徴とする。

50

【 0 0 4 8 】

また、上記課題を解決するために、本発明は、前記更新結果情報生成段階では、過去に行った更新の更新結果情報に基づく更新履歴情報に、生成した更新結果情報を追記することを特徴とする。

【 0 0 4 9 】

また、上記課題を解決するために、本発明は、前記更新結果情報生成段階の後に、前記画像形成装置の再起動後を行う再起動段階をさらに有することを特徴とする。

【 0 0 5 0 】

また、上記課題を解決するために、本発明は、前記更新結果情報生成段階は、前記画像形成装置が再起動する前に、前記画像形成装置が有する不揮発性記憶部に、前記更新結果情報を記憶することを特徴とする。

10

【 発明の効果 】

【 0 0 5 1 】

以上説明したように、本発明によれば、プログラムをネットワークを介して更新する画像形成装置に最適なプログラム更新方法、そのプログラム更新方法を搭載した画像形成装置が得られる。

【 発明を実施するための最良の形態 】

【 0 0 5 2 】

以下、本発明の実施例を図面に基づいて説明する。なお、本実施例において、更新プログラムをダウンロードするとは、更新プログラムを取得し、メモリなどの所定の記憶領域に更新プログラムを格納することとする。また、更新プログラムとは、現在運用されているプログラムに代わる新たなプログラムであり、更新用プログラムとは、運用されているプログラムを更新プログラムに更新する処理を実行するプログラムである。

20

【 実施例 】

【 0 0 5 3 】

図 1 は、本発明による融合機の一実施例の構成図を示す。融合機 1 は、ソフトウェア群 2 と、融合機起動部 3 と、ハードウェア資源 4 とを含むように構成される。

【 0 0 5 4 】

融合機起動部 3 は融合機 1 の電源投入時に最初に実行され、アプリケーション層 5 およびプラットフォーム層 6 を起動する。例えば融合機起動部 3 は、アプリケーション層 5 およびプラットフォーム層 6 のプログラムを、ハードディスク装置（以下、HDDという）などから読み出し、読み出した各プログラムをメモリ領域に転送して起動する。ハードウェア資源 4 は、スキャナ 2 5 と、プロッタ 2 6 と、ADF（Auto Document Feeder）などのハードウェアリソース 2 4 とを含む。

30

【 0 0 5 5 】

また、ソフトウェア群 2 は、UNIX（登録商標）などのオペレーティングシステム（以下、OSという）上に起動されているアプリケーション層 5 とプラットフォーム層 6 とを含む。アプリケーション層 5 は、プリンタ、コピー、ファックスおよびスキャナなどの画像形成にかかるユーザサービスにそれぞれ固有の処理を行うプログラムを含む。

【 0 0 5 6 】

アプリケーション層 5 は、プリンタ用のアプリケーションであるプリンタアプリ 9 と、コピー用アプリケーションであるコピーアプリ 1 0 と、ファックス用アプリケーションであるファックスアプリ 1 1 と、スキャナ用アプリケーションであるスキャナアプリ 1 2 とを含む。

40

【 0 0 5 7 】

また、プラットフォーム層 6 は、アプリケーション層 5 からの処理要求を解釈してハードウェア資源 4 の獲得要求を発生するコントロールサービス層 7 と、1 つ以上のハードウェア資源 4 の管理を行ってコントロールサービス層 7 からの獲得要求を調停するシステムリソースマネージャ（以下、SRMという）2 1 と、SRM 2 1 からの獲得要求に応じてハードウェア資源 4 の管理を行うハンドラ層 8 とを含む。

50

【 0 0 5 8 】

コントロールサービス層 7 は、ネットワークコントロールサービス（以下、NCS という）13、リモートサービス（以下、RS という）14、オペレーションパネルコントロールサービス（以下、OCS という）15、ファックスコントロールサービス（以下、FCS という）16、エンジンコントロールサービス（以下、ECS という）17、メモリコントロールサービス（以下、MCS という）18、オンデマンドアップデートサービス（以下、OUS という）19、システムコントロールサービス（以下、SCS という）20 など、一つ以上のサービスモジュールを含むように構成されている。

【 0 0 5 9 】

なお、プラットフォーム層 6 は予め定義されている関数により、アプリケーション層 5 からの処理要求を受信可能とする API 28 を有するように構成されている。OS は、アプリケーション層 5 およびプラットフォーム層 6 の各ソフトウェアをプロセスとして並列実行する。

10

【 0 0 6 0 】

通信部に対応する NCS 13 のプロセスは、ネットワーク I/O を必要とするアプリケーションに対して共通に利用できるサービスを提供するものであり、ネットワーク側から各プロトコルによって受信したデータを各アプリケーションに振り分けたり、各アプリケーションからのデータをネットワーク側に送信する際の仲介を行う。

【 0 0 6 1 】

例えば NCS 13 は、ネットワークを介して接続されるネットワーク機器とのデータ通信を httpd (HyperText Transfer Protocol Daemon) により、HTTP (HyperText Transfer Protocol) で制御する。

20

【 0 0 6 2 】

RS 14 のプロセスは、インターネットなどのネットワークを用いたサービスを行う。OCS 33 のプロセスは、オペレータと本体制御との間の情報伝達手段となるオペレーションパネルの制御を行う。FCS 16 のプロセスは、アプリケーション層 5 から PSTN または ISDN 網を利用したファックス送受信、バックアップ用のメモリで管理されている各種ファックスデータの登録/引用、ファックス読み取り、ファックス受信印刷などを行うための API を提供する。

【 0 0 6 3 】

ECS 17 のプロセスは、スキャナ 25、プロッタ 26、その他のハードウェアリソース 24 などのエンジン部の制御を行う。MCS 18 のプロセスは、メモリの取得および解放、HDD の利用などのメモリ制御を行う。更新プログラム取得部に対応する OUS 19 は、ネットワークからの通知により、プログラムをダウンロードし、メモリに格納する。

30

【 0 0 6 4 】

プログラム更新部、更新結果情報生成部、更新結果出力部、再起動部に対応する SCS 20 のプロセスは、アプリケーション管理、操作部制御、システム画面表示、LED 表示、ハードウェア資源管理、割り込みアプリケーション制御などの処理を行う。

【 0 0 6 5 】

SRM 21 のプロセスは、SCS 20 と共にシステムの制御およびハードウェア資源 4 の管理を行うものである。例えば SRM 21 のプロセスは、スキャナ 25 やプロッタ 26 などのハードウェア資源 4 を利用する上位層からの獲得要求に従って調停を行い、実行制御する。

40

【 0 0 6 6 】

具体的に、SRM 21 のプロセスは獲得要求されたハードウェア資源 4 が利用可能であるか（他の獲得要求により利用されていないかどうか）を判定し、利用可能であれば獲得要求されたハードウェア資源 4 が利用可能である旨を上位層に通知する。また、SRM 21 のプロセスは上位層からの獲得要求に対してハードウェア資源 4 を利用するためのスケジューリングを行い、要求内容（例えば、プリンタエンジンによる紙搬送と作像動作、メモリ確保、ファイル生成など）を直接実施している。

50

【 0 0 6 7 】

また、ハンドラ層 8 は後述するファックスコントロールユニット（以下、FCU という）の管理を行うファックスコントロールユニットハンドラ（以下、FCUH という）22 と、プロセスに対するメモリの割り振り及びプロセスに割り振ったメモリの管理を行うイメージメモリハンドラ（以下、IMH という）23 を含む。SRM39 および FCUH22 は、予め定義されている関数によりハードウェア資源 4 に対する処理要求を送信可能とするエンジン I/F27 を利用して、ハードウェア資源 4 に対する処理要求を行う。

【 0 0 6 8 】

融合機 1 は、各アプリケーションで共通的に必要な処理をプラットフォーム層 6 で一元的に処理することができる。次に、融合機 1 のハードウェア構成について説明する。

10

【 0 0 6 9 】

図 2 は、融合機 1 の一実施例のハードウェア構成図を示している。融合機 1 は、コントローラボード 50 と、オペレーションパネル 39 と、FCU40 と、エンジン 43 とを含む。また、FCU40 は、G3 規格対応ユニット 103 と、G4 規格対応ユニット 104 とを有する。

【 0 0 7 0 】

また、コントローラボード 50 は、CPU31 と、ASIC36 と、HDD38 と、システムメモリ（MEM-P）32 と、ローカルメモリ（MEM-C）37 と、ノースブリッジ（以下、NB と記す）33 と、サウスブリッジ（以下、SB と記す）34 と、NIC101（Network Interface Card）と、USB デバイス 41 と、IEEE1394 デバイス 42 と、セントロニクスデバイス 102 と、MLB45 とを含む。

20

【 0 0 7 1 】

オペレーションパネル 39 は、コントローラボード 50 の ASIC36 に接続されている。また、SB34 と、NIC101 と、USB デバイス 41 と、IEEE1394 デバイス 42 と、セントロニクスデバイス 102 と、NB33 に PCI バスで接続されている。

【 0 0 7 2 】

MLB45 は、融合機 1 に PCI バスを介して接続する基板である。また、MLB45 は、融合機 1 から入力された画像データを変換し、変換された画像データあるいは符号化された画像データを融合機 1 に出力するものである。

30

【 0 0 7 3 】

また、FCU40 と、エンジン 43 は、コントローラボード 50 の ASIC36 に PCI バスで接続されている。

【 0 0 7 4 】

なお、コントローラボード 50 は、ASIC36 にローカルメモリ 37、HDD38 などが接続されると共に、CPU31 と ASIC36 とが CPU チップセットの NB33 を介して接続されている。このように、NB33 を介して CPU31 と ASIC36 とを接続すれば、CPU31 のインタフェースが公開されていない場合に対応できる。

【 0 0 7 5 】

また、ASIC36 と NB33 とは PCI バスを介して接続されているのではなく、AGP（Accelerated Graphics Port）67 を介して接続されている。このように、図 2 のアプリケーション層 5 やプラットフォーム層 6 を形成する一つ以上のプロセスを実行制御するため、ASIC36 と NB33 とを低速の PCI バスでなく AGP35 を介して接続し、パフォーマンスの低下を防いでいる。

40

【 0 0 7 6 】

CPU31 は、融合機 1 の全体制御を行うものである。CPU31 は、NCS13、RS14、OCS15、FCS16、ECS17、MCS18、OUS19、SCS20、SRM21、FCUH22、IMH23、MEU44 を OS 上にそれぞれプロセスとして起動して実行させると共に、アプリケーション層 5 を形成するプリンタアプリ 9、コピーアプリ 10、ファックスアプリ 11、スキャナアプリ 12 を起動して実行させる。

50

【 0 0 7 7 】

N B 3 3 は、C P U 3 1、システムメモリ 3 2、S B 3 4 および A S I C 3 6 を接続するためのブリッジである。システムメモリ 3 2 は、融合機 1 の描画用メモリなどとして用いるメモリである。S B 3 4 は、N B 3 3 と P C I バス、周辺デバイスとを接続するためのブリッジである。また、ローカルメモリ 3 7 はコピー用画像バッファ、符号バッファとして用いるメモリである。

【 0 0 7 8 】

A S I C 3 6 は、画像処理用のハードウェア要素を有する画像処理用途向けの I C である。H D D 3 8 は、画像の蓄積、プログラムの蓄積、フォントデータの蓄積、フォームの蓄積などを行うためのストレージである。また、オペレーションパネル 3 9 は、ユーザからの入力操作を受け付けると共に、ユーザに向けた表示を行う操作部である。

10

【 0 0 7 9 】

次に、図 3 を用いて、各処理部とプログラムの融合機 1 における位置づけを説明する。図 3 は、各処理部と、プログラムであるファームウェアの位置づけを示す図である。図 3 には、制御部であるコントローラボード 5 0 と、処理部である操作パネルボード 5 5、F C U ボード 5 1、エンジンボード 5 2、フィニッシャ 5 3、A D F 5 4 とが示されている。また、制御部のファームウェアであるソフトウェア群 2 と、操作パネルボード 5 5 のファームウェアである操作パネルファーム 6 1 と、F C U ボード 5 1 のファームウェアである F C U ファーム 5 6 と、エンジンボード 5 2 のファームウェアであるプロッタファーム 5 7 とスキャナファーム 5 8 と、フィニッシャ 5 3 のファームウェアであるフィニッシャ

20

【 0 0 8 0 】

まずコントローラボード 5 0 について説明する。コントローラボード 5 0 は、図 1 に示したソフトウェア群 2 が実行されるボードで、融合機 1 を制御する。このコントローラボード 5 0 から、以下に説明するボードが接続される。

【 0 0 8 1 】

操作パネルボード 5 5 は、オペレーションパネル 3 9 の制御を行うボードであり、融合機 1 に関する情報を、融合機 1 を操作するオペレータに対して表示する処理を行う。この操作パネルボード 5 5 とコントローラボード 5 0 とは、通信をするために同期シリアル回線 6 4 で接続されている。

30

【 0 0 8 2 】

次に、F C U ボード 5 1 について説明する。このボードは、上述したファックスコントロールユニットである。また、F C U ボード 5 1 とコントローラボード 5 0 とは、通信をするために P C I バス 6 2 で接続される。

【 0 0 8 3 】

次に、エンジンボード 5 2 について説明する。エンジンボード 5 2 は、プロッタファーム 5 7 やスキャナファーム 5 8 などのエンジン部のファームが搭載される。このエンジンボード 5 2 もコントローラボード 5 0 と通信をするために P C I バス 6 2 で接続される。

【 0 0 8 4 】

以上説明した操作パネルボード 5 5 と、F C U ボード 5 1 と、エンジンボード 5 2 とは、それぞれコントローラボード 5 0 と直接通信するボードである。

40

【 0 0 8 5 】

次に説明するフィニッシャ 5 3 と A D F は、エンジンボード 5 2 とシリアル回線で通信することにより、エンジンボード 5 2 を介してコントローラボード 5 0 と通信するボードである。

【 0 0 8 6 】

フィニッシャ 5 3 は、印刷用紙をステーブラでとじるための装置であり、フィニッシャファーム 5 9 により動作する。A D F 5 4 は、自動で原稿を送る装置であり、A D F ファーム 6 0 により動作する。

50

【 0 0 8 7 】

以上説明した操作パネルボード55と、FCUボード51と、エンジンボード52と、フィニッシャ53と、ADF54は、いずれのボードもそれぞれCPUを有する。

【 0 0 8 8 】

このように融合機1は、画像形成に係る処理を行う1つ以上の処理部と、処理部と直接または他の処理部を介して通信する制御部と、処理部と制御部とを動作させるプログラムとを有する。

【 0 0 8 9 】

次に、各ボードの立ち上げ時の処理を、図4を用いて説明する。図4は、コントローラボード50と、エンジンボード52と、FCUH22と、OCS19との間のシーケンス図を示している。なお、FCUH22は、FCUボード62の立ち上げを管理し、OCS19は、操作パネルボード55の立ち上げを管理している。

10

【 0 0 9 0 】

まず、ステップS1とステップS2で、コントローラボード50及びエンジンボード52は、互いに通信可能なことを確認する。そして、コントローラボード50は、ステップS3で、エンジンボード52に対し、コントローラと接続されたことを通知する。エンジンボード52も同様に、ステップS3で、コントローラボードに対し、エンジンが接続したことを通知する。このとき、エンジンボード52は、搭載されているプログラムに関する情報も同時に送信する。

【 0 0 9 1 】

FCUH22は、ステップS5で、FCUボード51が起動したことを確認すると、FCUボード51が起動したことと、FCUボード51に搭載されているプログラムに関する情報をコントローラボード50に送信する。また、OCS19も同様に、操作パネルボード55が起動すると、OCSが起動したことと、操作パネルボード55に搭載されているプログラムに関する情報を送信する。

20

【 0 0 9 2 】

次に、プログラムを更新するときの処理について説明する。

【 0 0 9 3 】

上記処理部のプログラムの更新は、コントローラボード50のSCS20の指示により行われる。まず、このSCS20が含まれるコントローラボード50に搭載されているプログラムの更新について説明する。

30

【 0 0 9 4 】

コントローラボード50に搭載されている各アプリ9、10、11、12のプログラムと、コントローラ6のプログラム(図1参照)であるが、それらは、個々に更新を行えるようになっている。

【 0 0 9 5 】

また、コントローラ6の各プログラムは、通常の処理をしている場合、NV-RAM47に書き込まれたプログラムで動作している。したがって、プログラムの更新をする場合は、NV-RAM47上のプログラムを書き換えることはできないため、実行しているプログラムをローカルメモリ37あるいはシステムメモリ32などのRAMに展開し、その展開したプログラムで処理を実行するようになる。

40

【 0 0 9 6 】

しかし、RAMは、多くの容量を持たないため、コントローラボード50は、通常の動作を実行するためのプログラムとは別に、プログラムを更新するための更新用プログラムを用意しておき、その更新用プログラムをRAMに展開してプログラムの更新を行う。図5は、NV-RAMからRAMへの更新用プログラムの展開を示す図である。図5に示されるように、NV-RAMに書き込まれている更新用プログラム201が、RAMに更新用プログラム202として展開される。

【 0 0 9 7 】

この更新用プログラムと通常の動作を実行するためのプログラムの違いについて説明す

50

る。

【 0 0 9 8 】

コントローラ 6 のプログラムは、多くの処理部と通信を行い、融合機 1 の制御を行うため、通信を行うためのプログラムを含んでいる。図 6 は、コントローラプログラム 2 0 3 と、周辺ソフトプログラム 2 0 4 とが通信可能な様子を示すものである。なお、図 6 における周辺ソフトプログラムとは、図 3 に示される F C U ファーム 5 6 など直接に通信を行うソフトでもよいし、エンジンボード 5 2 に搭載されているファームでもよい。

【 0 0 9 9 】

このようにコントローラボード 5 0 は、他の処理部の制御を行うため、送受信する内容を実行するためのプログラムは、サイズが大きいプログラムである。

10

【 0 1 0 0 】

更新用プログラムは、上記通信を行うためのプログラムを除くことにより、サイズの小さいプログラムとなったプログラムに、更新する処理を加えたプログラムであるので、通常の処理を実行するためのプログラムと比較して小さなサイズのプログラムである。

【 0 1 0 1 】

この更新用プログラムによって、周辺ソフトとの通信は行えなくなるが、R A M を占有する領域を減らすことが可能となる。図 7 は、更新用プログラム 2 0 3 の実行中の状態における通信の様子を示すもので、コントローラボード 5 0 と周辺ソフトプログラム 2 0 4 とが通信を行えなくなった様子を示す

このように、コントローラボード 5 0 に搭載されたコントローラ 6 のプログラムを更新すると、各処理部との通信が行えなくなる。したがって、この場合、コントローラ 6 のプログラムは、他の処理部のプログラムの更新を行えないことになる。

20

【 0 1 0 2 】

このことは、フィニッシャ 5 3 あるいは A D F 5 4 と通信するエンジンボード 5 2 についても言えることである。フィニッシャ 5 3 あるいは A D F 5 4 は、エンジンボード 5 2 からプログラムの更新を指示されることになるため、エンジンボード 5 2 のプログラムを更新すると、エンジンボード 5 2 は、コントローラボード 5 0 と同様にフィニッシャ 5 3 あるいは A D F 5 4 との通信が行えなくなり、フィニッシャ 5 3 あるいは A D F 5 4 のプログラムの更新を行えないことになる。

【 0 1 0 3 】

すなわち、コントローラボード 5 0 と通信するために経由する処理部の数が多いものほど、先にプログラムの更新を行わなければならない。

30

【 0 1 0 4 】

そこで次に、プログラムを更新する順番を定めるための優先度情報について図 8 を用いて説明する。図 8 に示される優先度情報は、プログラム種類とそのプログラムの優先度を示すテーブルである。プログラム種類とは、F C U ファームやフィニッシャファーム、プリンタアプリなどを表す。

【 0 1 0 5 】

また、優先度であるが、例えば、プログラム C の優先度は 2 であり、プログラム E の優先度は 1 0 である。この優先度の値が小さい方が更新する順番が先になる。したがって、プログラム D よりもプログラム A の方が更新する順番が先になる。また、同じ順位の場合は、その順位のいずれのプログラムから更新しても良い。

40

【 0 1 0 6 】

この優先度情報は、各処理部とコントローラボード 5 0 との通信をするために経由する処理部の数に応じて定まる。したがって、図 3 に示されるような処理部の配置であれば、エンジンボード 5 2 を経由してコントローラボード 5 0 と通信を行うフィニッシャファーム 5 9 と A D F ファーム 6 0 の優先度が 1 となる。

【 0 1 0 7 】

次に、この優先度情報に応じて取得した更新プログラムで更新する順番を定める更新順番情報と、その生成する過程について説明する。

50

【 0 1 0 8 】

図 9 は、更新順番情報を生成する過程を示した図である。更新プログラムをプログラム E、F、D、A とし、この順序でダウンロードされたとする。

【 0 1 0 9 】

コントローラ 6 は、優先度情報から取得した更新プログラムの優先度を参照し、取得した更新プログラムとその優先度の組で構成される優先度表 7 0 を生成する。そして、この優先度表 7 0 を優先度で昇順にソートし、更新順番情報 7 1 を生成する。また、更新順番情報 7 1 は、順番と取得した更新プログラムにより更新されるプログラムの種類とを対応付けた情報を有している。

【 0 1 1 0 】

このように、更新順番情報 7 1 を生成した後、更新順番情報 7 1 を拡張しても良い。図 1 0 は、更新順番情報 7 1 を拡張したもので、図に示されるようにバージョンや、実行状態を加えたものである。なお、図 1 0 の実行状態には、「未実行」、「実行中」、「実行済み」が示される。「未実行」は、プログラムの更新を行っていない状態を示す。「実行中」は、プログラムの更新を実行中である状態を示す、また、「実行済み」は、プログラムの更新の実行が済んだ状態を示す。以上のように、実行状態は、更新プログラムにより更新されるプログラムの更新状況を示す情報である。

【 0 1 1 1 】

なお、図 1 0 において、コントローラ 6 のプログラムの更新より操作パネルファームの更新の方が、後に行われるようになっている。その理由は、操作パネルファームがオペレータに対して進捗状況を描画するためである。そのため、操作パネルファームを更新する順番が最後となる。図 1 1 は、操作パネルファーム 2 0 5 が書き込まれた N V - R A M から R A M 上に展開された更新プログラム 2 0 6 を示すものである。

【 0 1 1 2 】

操作パネルファームの更新をコントローラ 6 のプログラムの更新の後に行う場合、コントローラ 6 の更新用プログラムは、操作パネルボードとの通信を行う処理を含む。なお、進捗状況を描画する必要が無ければ、操作パネルファームを更新する順番も、他の処理部のプログラムの更新と同様に定める。

【 0 1 1 3 】

以上説明した更新順番情報は、コントローラ 6 の S C S 2 0 が保有する。図 1 2 は、更新順番情報のコントローラ 6 における位置を示す図である。図に示されるように更新順番情報 2 0 7 は、コントローラ 6 が保有する。

【 0 1 1 4 】

次に、融合機 1 におけるプログラムの更新処理を図 1 3 のフローチャートを用いて説明する。

【 0 1 1 5 】

更新プログラム取得段階に対応するステップ S 1 0 1 で、コントローラ 6 の S C S 2 0 は、ネットワークを介して更新プログラムを取得した O U S 1 9 からプログラム更新開始要求を受信する。次のステップ S 1 0 2 で、コントローラ 6 は、優先情報と R A M に展開された更新プログラムに応じて、更新順番情報を生成する。

【 0 1 1 6 】

次に、コントローラ 6 は、プログラム更新段階に対応するステップ S 1 0 3 で、更新順番情報に示される更新未実行状態のプログラムのうち、最優先のプログラムの更新を実行する。そして、コントローラ 6 は、ステップ S 1 0 4 で更新を実行しているプログラムの更新順番情報の状態を実行中に変更し、変更終了後実行済みに変更する。

【 0 1 1 7 】

次に、コントローラ 6 は、ステップ S 1 0 5 で、更新順番情報に登録されている全プログラムの更新が実行完了したかどうか判断し、完了していない場合は、再びステップ S 1 0 3 の処理を実行する。完了した場合、コントローラ 6 は、ステップ S 1 0 6 で更新したプログラムが搭載されるボードのリポートを実行し、処理を終了する。

10

20

30

40

50

【 0 1 1 8 】

次に、リポート処理について説明する。まず、ボードの種類について説明する。ボードには、コントローラ 6 が直接リポート可能なものと、コントローラ 6 がリポート実行要求をすることにより、ボードが自らリポートするものの 2 種類ある。これらのボードのリポート処理について説明する。

【 0 1 1 9 】

最初に、コントローラ 6 が直接にリポート可能なボードのリポート処理について、図 14 を用いて説明する。コントローラ 6 は、ステップ S 2 0 1 で、全ての更新対象プログラムの更新処理が完了すると、ステップ S 2 0 2 で、エンジンファームに対し、リポートしても不具合が起きないように、処理の実行を制限させる動作停止要求を通知する。

10

【 0 1 2 0 】

ステップ S 2 0 3 は、動作停止要求に対するエンジンファームからの動作停止完了待ちである。このステップで、動作停止完了を受信すると、コントローラ 6 はステップ S 2 0 4 で、操作パネルファームに動作停止要求を通知する。そして、ステップ S 2 0 5 で、コントローラ 6 は、再び動作停止要求に対する操作パネルファームからの動作停止完了待ちとなる。

【 0 1 2 1 】

このステップで動作停止完了を受信すると、コントローラ 6 はステップ S 2 0 6 で、リポートを実行する。

【 0 1 2 2 】

このように、コントローラ 6 が直接にリポート可能なボードの場合、コントローラ 6 は、リポートする前に動作停止要求を行う。

20

【 0 1 2 3 】

次に、コントローラ 6 がリポート実行要求をすることにより、ボードが自らリポートするボードのリポート処理について、図 15 を用いて説明する。コントローラ 6 は、ステップ S 3 0 1 で、全ての更新対象プログラムの更新処理が完了すると、ステップ S 3 0 2 で、エンジンファームに対し、リポート実行要求を通知する。次に、コントローラ 6 は、ステップ S 3 0 3 で、操作パネルファームにリポート実行要求を通知する。その一定時間後に、コントローラ 6 は、ステップ S 3 0 4 で、自らのリポートを実行する。そして、再起動後、コントローラ 6 は、ステップ S 3 0 5 で各処理部と通信を再開する。

30

【 0 1 2 4 】

このように、コントローラ 6 がリポート実行要求をすることにより、ボードが自らリポートするボードの場合、コントローラ 6 は、リポート実行要求を通知し、一定時間後に自らをリポートする。

【 0 1 2 5 】

上述したフローチャートの処理を、図 7 のシーケンス図を用いて説明する。このシーケンス図は、O U S 1 9 と、S C S 2 0 と、操作パネルファーム 6 1 と、システムソフト 7 2 と、エンジンファーム 7 3 との間で行われるやり取りを示すものである。なお、システムソフト 7 2 は、図 1 に示されるコントローラ 6 とアプリケーション層 5 のプログラムを表す。エンジンファーム 7 3 は、エンジンボードに搭載されているプロッタファーム 5 7 あるいはスキャナファーム 5 8 を表す。

40

【 0 1 2 6 】

更新プログラム取得段階に対応するステップ S 4 0 1 で、コントローラ 6 の O U S 1 9 は、同じコントローラ 6 の S C S 2 0 に対し、更新プログラムをダウンロードしたことを通知する。この通知により、S C S 2 0 は、モジュール判定処理を行い、後述する更新管理テーブルを生成する。更新管理テーブルを生成した S C S 2 0 は、プログラム更新段階に対応するステップ S 4 0 2 でエンジンファーム 7 3 に更新プログラムを送信する。エンジンファーム 7 3 は、プログラムの更新が完了するとステップ S 4 0 3 で、S C S 2 0 に更新結果応答を通知する。

【 0 1 2 7 】

50

次に、SCS20は、ステップS404で、コントローラ6のプログラムを更新するために、SCS20を除くシステムソフト72にリポート準備要求を通知する。システムソフト72は、リポートの準備が完了すると、ステップS405で、リポート準備完了を通知する。

【0128】

次に、SCS20は、揮発性記憶部であるRAM実行へ切り替える(図5参照)。RAM実行に切り替わったSCS20は、ステップS406でシステムソフト72のプログラムの更新処理を行う。次に、SCS20は、ステップS407で、操作パネルファーム61に対し、更新プログラムを送信する。操作パネルファーム61は、ステップS408で、SCS20に対しプログラムを更新した結果を通知する。

10

【0129】

SCS20は、全てのプログラムの更新が完了したため、再起動段階に対応するステップS409でエンジンファーム73と、操作パネルファーム71とをステップS409でリポートする。そして、SCS20は、t秒後にシステムソフト72をリポートする。

【0130】

上述した更新管理テーブルの生成について、図17を用いて説明する。この更新管理テーブルは、テーブルの項目として、プログラム種類と、そのプログラムのバージョンと、受信プログラムデータのモジュールIDと、更新の実行状態とを有する。

【0131】

受信プログラムデータのモジュールIDとは、プログラムデータを一意的に特定するものである。また、実行状態は、図10で説明した実行状態と同じものである。

20

【0132】

この更新実行用データテーブルにおいて、例えばスキャナエンジンプログラムは、バージョンが1.22で、モジュールIDがJP2CFa_escanで、実行状態は未実行である。

【0133】

この更新管理テーブルの生成を説明するため、まず、図18に示されるテーブルについて説明する。図18に示されるテーブルは、融合機1に搭載されているモジュールIDの構成を示すものである。

【0134】

このテーブルは、項目として、融合機1のプログラムのプログラム種類と、そのプログラムのバージョンと、モジュールIDとを有する。融合機1の構成が、図18に示される構成であるとする。また、ダウンロードした更新プログラムが、プロッタエンジンプログラム、スキャナエンジンプログラム、コントローラプログラム、プリンタアプリプログラム、FAXアプリプログラム、操作部プログラムの6つであるとする。

30

【0135】

このときSCS20は、当該機種 of プログラム構成情報に含まれないプログラムを外し、図19に示される更新管理テーブルを生成する。今の場合、融合機1の構成にFAXアプリプログラムが搭載されていないため、図19に示されるように、FAXアプリプログラムを除いた更新管理テーブルが生成される。

【0136】

次に、SCS2がプログラムを更新した順にリポートさせる処理について、図20のフローチャートを用いて説明する。

40

【0137】

プログラム更新段階に対応するステップS501で、SCS20は、エンジンファーム73のプログラム更新処理を実行する。次のステップS502で、SCS20は、エンジンファーム73に動作停止要求を通知する。

【0138】

ステップS503は、エンジンファーム73から送信される動作停止完了の受信待ちである。この動作停止完了をSCS20が受信すると、再起動段階に対応するステップS504で、SCS20は、エンジンファーム73のリポートを実行し、ステップS512へ

50

処理を進める。

【0139】

ステップS502の処理に戻る。ステップS502で、エンジンファーム73に動作停止要求を通知したSCS20は、ステップS505で、システムソフト72のプログラム更新処理を実行する。次のステップS506で、SCS20は、SCS20以外のシステムソフト72に動作停止要求を通知する。

【0140】

ステップS507は、システムソフト72から送信される動作停止完了の受信待ちである。この動作停止完了をSCS20が受信すると、SCS20は、ステップS512へ処理を進める。

10

【0141】

ステップS506の処理に戻る。ステップS506で、システムソフト72に動作停止要求を通知したSCS20は、ステップS508で、操作パネルファーム61のプログラム更新処理を実行する。次のステップS509で、SCS20は、操作パネルファーム61に動作停止要求を通知する。

【0142】

ステップS510は、操作パネルファーム61から送信される動作停止完了の受信待ちである。この動作停止完了をSCS20が受信すると、SCS20は、再起動段階に対応するステップS511で操作パネルファーム61のリポートを実行する。

【0143】

SCS20は、ステップS512で、エンジンファーム73と操作パネルファーム61のリポートが完了したと判断すると、ステップS513でシステムソフト72のリポートを実行し処理を終了する。

20

【0144】

次に、上述した処理を、図21のシーケンス図を用いて説明する。このシーケンス図は、先ほどのシーケンス図と同様に、OUS19と、SCS20と、操作パネルファーム61と、システムソフト72と、エンジンファーム73との間で行われるやり取りを示すものである。

【0145】

更新プログラム取得段階に対応するステップS601で、OUS19は、SCS20に対し、更新プログラムをダウンロードしたことを通知する。この通知により、SCS20は、上述したモジュール判定処理を行い、更新管理テーブルを生成する。

30

【0146】

次に、SCS20は、プログラム更新段階に対応するステップS602でエンジンファーム73に更新プログラムを送信する。エンジンファーム73は、プログラムの更新が完了すると、ステップS603で、SCS20に更新結果応答を通知する。更新結果応答を通知されたSCS20は、再起動段階に対応するステップS604で、エンジンファーム73に対し、リポート実行要求を行い、エンジンファーム73は、リポートを実行する。

【0147】

次に、SCS20は、ステップS605で、コントローラ6のプログラムを更新するために、SCS20を除くシステムソフト72にリポート準備要求を通知する。システムソフト72は、リポートの準備が完了すると、ステップS606で、リポート準備完了を通知する。そして、SCS20は、上述したRAM実行へ切り替える。

40

【0148】

RAM実行に切り替わったSCS20は、ステップS607でシステムソフトのプログラムの更新処理を行う。次に、SCS20は、ステップS608で、操作パネルファーム61に対し、更新プログラムを送信する。操作パネルファーム61は、ステップS609で、SCS20に対しプログラムを更新した結果を通知する。

【0149】

次に、SCS20は、再起動段階に対応するステップS610で、操作パネルファーム

50

に対し、レポート実行要求を行い、操作パネルファーム 6 1 は、レポートを実行する。

【 0 1 5 0 】

そして、SCS 2 0 は、全てのプログラムの更新が完了したため、t 秒後にシステムソフト 7 2 をレポートする。

【 0 1 5 1 】

次に、SCS 2 0 が、操作パネルファーム 6 1 やエンジンファーム 7 3 との通信回線が切断されたことを検出することで、SCS 2 0 が自らのレポートを実行する処理について説明する。

【 0 1 5 2 】

通信回線が切断されることは、通信相手がレポートしたことを意味する。したがって、この処理は、SCS 2 0 の通信相手が確実にレポートを実行したかどうか確認した後に、SCS 2 0 が自らのレポートを実行する処理である。

10

【 0 1 5 3 】

この処理を、図 2 2 を用いて説明する。更新プログラム取得段階に対応するステップ S 7 0 1 で、OUS 1 9 は、SCS 2 0 に対し、更新プログラムをダウンロードしたことを通知する。この通知により、SCS 2 0 は、モジュール判定処理を行い、上述した更新管理テーブルを生成する。

【 0 1 5 4 】

更新管理テーブルを生成した SCS 2 0 は、プログラム更新段階に対応するステップ S 7 0 2 でエンジンファーム 7 3 に更新プログラムを送信する。エンジンファーム 7 3 は、プログラムの更新が完了するとステップ S 7 0 3 で、SCS 2 0 に更新結果応答を通知する。

20

【 0 1 5 5 】

次に、SCS 2 0 は、ステップ S 7 0 4 で、コントローラ 6 のプログラムを更新するために、SCS 2 0 を除くシステムソフト 7 2 にレポート準備要求を通知する。システムソフト 7 2 は、レポートの準備が完了すると、ステップ S 7 0 5 で、レポート準備完了を通知する。

【 0 1 5 6 】

次に、SCS 2 0 は、RAM 実行へ切り替える。RAM 実行に切り替わった SCS 2 0 は、ステップ S 7 0 6 でシステムソフト 7 2 のプログラムの更新処理を行う。次に、SCS 2 0 は、ステップ S 7 0 7 で、操作パネルファーム 6 1 に対し、更新プログラムを送信する。操作パネルファーム 6 1 は、ステップ S 7 0 8 で、SCS 2 0 に対しプログラムを更新した結果を通知する。

30

【 0 1 5 7 】

そして、SCS 2 0 は、全てのプログラムの更新が完了したため、再起動段階に対応するステップ S 7 0 9 でエンジンファーム 7 3 にレポート実行要求を行う。これにより、ステップ S 7 1 0 で、エンジンファーム 7 3 との通信回線が切断されることが確認される。

【 0 1 5 8 】

次に、SCS 2 0 は、再起動段階に対応するステップ S 7 1 1 で操作パネルファーム 6 1 にレポート実行要求を行う。これにより、ステップ S 7 1 2 で、操作パネルファーム 6 1 との通信回線が切断されることが確認される。

40

【 0 1 5 9 】

次に、SCS 2 0 は、エンジンファーム 7 3 と操作パネル 6 1 がレポートを実行したことが確認できたので、SCS 2 0 は、システムソフト 7 2 をレポートする。

【 0 1 6 0 】

以上、レポートする方法として、全てのプログラムの更新が完了してからレポートする方法と、プログラムの更新が完了した順にレポートする方法の 2 つの方法について説明した。

【 0 1 6 1 】

これら 2 つの方法は、融合機 1 がレポートするまでの時間が異なっている。図 2 3 は、

50

全ての処理部のプログラムの更新が終了した後に、リポートする場合に要する時間を示すものである。

【0162】

図23には、プログラムの更新をFCUファームから行き、そのFCUファームのプログラムの更新が終了した時刻aに、エンジンファームのプログラムの更新を開始していることが示されている。また、エンジンファームのプログラムの更新処理が終了した時刻bに操作パネルファームの更新が開始することが示されている。そして、操作パネルファームのプログラムの更新が終了した時刻cにリポートを実行し、時刻dにリポートが終了したことが示されている。

【0163】

次の図24は、プログラムの更新が完了した順にリポートする方法で行った場合に要する時間を示したものである。

【0164】

図24には、プログラムの更新がFCUファームから行われ、そのFCUファームのプログラムの更新が終了した時刻aに、エンジンファームのプログラムの更新とFCUファームのリポート処理とが開始されていることが示されている。また、エンジンファームのプログラムの更新処理が終了した時刻bに操作パネルファームの更新とエンジンファームのリポート処理とを開始していることが示されている。そして、操作パネルファームのプログラムの更新が終了した時間cに操作パネルファームのリポートを実行し、時間dにリポートが終了したことが示されている。

【0165】

このように、プログラムの更新とリポートを平行して行くと、処理時間が短縮することができる。また、平行して処理を行う場合は、リポートする時間が長くかかるプログラムの更新を早く行うほど時間を短縮することが可能となる。

【0166】

次に、図25を用いて、表示部であるオペレーションパネル39の表示処理について説明する。

【0167】

更新プログラム取得段階に対応するステップS801で、SCS20は、OUS19から更新プログラムをダウンロードしたことを通知される。次のステップS802で、SCS20は、更新管理テーブルを生成する。そして、SCS20は、ステップS803で、更新未実行状態のプログラムのうち、テーブルの最上位のものプログラムの更新を実行する。

【0168】

次に、SCS20は、ステップS804で、省エネモードかどうか判断し、省エネモードでなければ、ステップS807へ処理を進める。省エネモードの場合、SCS20は操作パネルファーム61に対してパネルを非表示にさせる。この間、SCS20は、ステップS806で、省エネモードを解除するイベントが起きているかどうか判断する。

【0169】

ここで、省エネモードの解除するイベントとは、例えば、タッチセンサを有するオペレーションパネル39が触れられたなどのイベントをいう。

【0170】

省エネモードが解除されると、ステップS807で、SCS20は、更新の進捗状況を表示し、処理を終了する。

【0171】

上述した省エネモードについて説明する。省エネモードには、待機状態、予熱状態、低電力状態、静音状態、エンジンオフ状態の5種類のモードが存在する。

【0172】

待機状態とは、印刷可能な状態である。予熱状態と低電力状態とは、定着温度を下げた状態で、低電力状態はエネルギースターに対応している。なお、定着温度を何度まで下げ

10

20

30

40

50

るか、あるいは下げないかは、機種や設定により異なる。静音状態とは、定着温度に係る電源の供給をやめ、周辺機器が静音起動し、HDDのアクセスが可能な状態であり、ファックスの着信やデータの入力に対応しているものである。エンジンオフ状態とは、エンジンを停止させた状態であり、例えばエンジンのCPUがスリープしている、もしくは電氣的に遮断されている状態である。

【0173】

次に、図26を用いて、融合機1の状態を保持し、プログラム更新後に融合機1を更新前の状態に移行させる処理について説明する。

【0174】

更新プログラム取得段階に対応するステップS901で、SCS20は、OUS19から更新プログラムをダウンロードしたことを通知される。次に、SCS20は、現在の融合機1の状態を不揮発性記憶部であるNV-RAM47に記憶する。

10

【0175】

なお、融合機1の状態とは、例えば先ほど省エネ状態などであり、オペレーションパネル39の画面を暗くしている状態、予熱状態、定着温度を抑える、エンジンを停止しているなどの状態が挙げられる。

【0176】

ステップS903で、プログラムの更新実行が完了すると、SCS20は、ステップS904でリポートを実行する。

【0177】

そして、再起動後のステップS904でSCS20は、NV-RAM47に記憶していた融合機1の状態に融合機を移行する。

20

【0178】

以下、利用者や管理者などからなるオペレータにプログラムの更新をしたことを通知する処理について説明する。図27に示されるフローチャートは、プログラム更新後、印刷用紙に更新結果を印刷することによりオペレータに通知するSCS20の処理を示している。

【0179】

このフローチャートの説明をする。更新プログラム取得段階に対応するステップS1001でSCS20は、OUS19からプログラム更新開始要求を受信する。次に、SCS20は、通知された更新プログラムから、更新管理テーブルを生成する。

30

【0180】

更新管理テーブルを生成したSCS20は、プログラム更新段階に対応するステップS1003で、更新管理テーブルの実行状態が未実行状態のプログラムのうち、最上位のものプログラムの更新を実行する。

【0181】

ステップS1004で、SCS20は、更新を実行したプログラムの更新管理テーブルにおける実行状態を実行中に変更する。次に、SCS20は、更新結果情報生成段階に対応するステップS1005で、NV-RAM47に記憶される更新フラグを更新したプログラムごとに書き込む。この書き込み内容については後述する。

40

【0182】

SCS20は、ステップS1006で更新実行用テーブルに登録されている全プログラムの更新が実行完了したかどうか判断し、完了していなければ、再びステップS1003の処理を行い、完了していればステップS1007へ処理を進める。

【0183】

ステップS1007で、SCS20は、後述する更新日時情報に更新日時を書き込み、リポートを実行する。そして、ステップS1008でリポート実行が完了、すなわち再起動後、SCS20は、ステップS1009で、NV-RAM47に更新されたプログラムの情報があるかどうか判断する。

【0184】

50

NV - RAM 47に更新されたプログラムの情報がない場合、SCS 20は、ステップ S 1 0 1 0でオペレーションパネル 39に通常の画面を表示し、処理を終了する。

【0185】

NV - RAM 47に更新されたプログラムの情報がある場合、SCS 20は、更新結果出力段階に対応するステップ S 1 0 1 1で、更新結果情報に基づいた更新結果をプリントし、排出する。このプリント内容については、後に説明する。

【0186】

このように、プリントし、排出した後、SCS 20は、ステップ S 1 0 1 2で、NV - RAM 47に記憶していた更新結果情報を消去し処理を終了する。

【0187】

以上説明したフローチャートにおけるステップ S 1 0 0 5で、NV - RAM 47に書き込まれる内容について、図 28を用いて説明する。

【0188】

図 28に示される表は、NV - RAM 47に書き込まれるデータ構造を示すものである。このデータ構造は、項目として、プログラム名と、更新前のバージョンと、更新後のバージョンと、更新結果を有する。

【0189】

このうち、プログラム名は、エンジンプログラムのように、プログラムの名前を表し、例えば、文字コードまたは予め割り振られたコードで表すことができる。更新前のバージョンは、そのプログラムの更新する前のバージョンを表し、同様に文字コードで表すことができる。更新後のバージョンは、そのプログラムを更新した後のバージョンを表し、同様に文字コードで表すことができる。プログラムの更新の成否を示す更新結果は、プログラムの更新が成功したかどうかを表すデータであり、このデータは、1ビット以上のデータであれば表すことができる。

【0190】

なお、図中のプリンタアプリプログラムの更新後と更新結果に示されている「NULL」は、そのデータが「0」で表されており、そのデータの内容がないことを示す。したがって、プリンタアプリプログラムは、更新されなかったことが示されている。

【0191】

次に、図 29を用いて、NV - RAM 47に書き込まれる更新日時情報について説明する。図 29に示されるように、更新日時情報は、西暦、月、日、時、分で書き込まれる。

【0192】

図 28、図 29に示したデータが更新結果情報であるが、オペレータに提供する更新結果に応じ、更新結果情報の内容を適宜変更しても良い。例えば単にプログラムの更新の成否だけ提供するのであれば、1ビット以上の情報があれば十分である。

【0193】

次に、ステップ S 1 0 1 1でプリントされる内容について、図 30を用いて説明する。プリントした内容は、更新結果情報が有するプログラム名 2 1 1と、更新前のバージョン 2 1 2と更新後のバージョン 2 1 3と、更新結果 2 1 4と、更新日時 2 1 5と、その他の定型文 2 1 6、2 1 7になっている。

【0194】

このように、プリントされる内容には、更新結果情報が有する情報がプリントされ、オペレータなどにとっては、必要な情報が揃ったものとなっている。

【0195】

次に、更新結果をオペレータに電子メールで送信する処理について、図 31のフローチャートを用いて説明する。なお、図 31のフローチャートのステップ S 1 1 0 1からステップ S 1 1 0 8までは、図 27のフローチャートのステップ S 1 0 0 1からステップ S 1 0 0 8と同じ処理のため、説明を省略し、ステップ S 1 1 0 9以降の処理について説明する。

【0196】

10

20

30

40

50

リポート実行完了後、SCS20は、ステップS1109で、NV-RAM47に更新されたプログラムの情報があるかどうか判断する。

【0197】

NV-RAM47に更新されたプログラムの情報がない場合、SCS20は、ステップS1110でオペレーションパネル39に通常の画面を表示し、処理を終了する。

【0198】

NV-RAM47に更新されたプログラムの情報がある場合、SCS20は、ステップS1111で、管理者のメールアドレスが設定されているかどうか判断する。管理者のメールアドレスが設定されていない場合、SCS20は、ステップS1113へ処理を進める。

【0199】

管理者のメールアドレスが設定されている場合、ステップS1112で、SCS20は、管理者宛に更新結果情報に基づいた更新結果が記された電子メールを送信する。電子メールを送信した後、SCS20は、ステップS213で、NV-RAM47に記憶していた更新結果情報を消去し処理を終了する。

【0200】

送信した電子メールの内容を、図32を用いて説明する。図32に示されるように、更新結果情報が有するプログラム名211と、更新前のバージョン212と更新後のバージョン213と、更新結果214と、更新日時215と、その他の定型文216、217になっている。電子メールで送信される内容も、プリントされる内容と同様のものとなる。

【0201】

次に、表示部であるオペレーションパネル39に表示することによってオペレータに通知する処理について、図33のフローチャートを用いて説明する。なお、図33のフローチャートのステップS1201からステップS1208までは、図27のフローチャートのステップS1001からステップS1008と同じ処理のため、説明を省略し、ステップS1209以降の処理について説明する。

【0202】

リポート実行完了後、SCS20は、ステップS1209で、NV-RAM47に更新されたプログラムの情報があるかどうか判断する。

【0203】

NV-RAM47に更新されたプログラムの情報がない場合、SCS20は、ステップS1210でオペレーションパネル39に通常の画面を表示し、処理を終了する。

【0204】

NV-RAM47に更新されたプログラムの情報がある場合、SCS20は、ステップS1211で、記憶した更新結果情報に基づいた更新結果をオペレーションパネル39に表示する。

【0205】

このように、オペレーションパネル39に表示した後、SCS20は、ステップS1212で、NV-RAM47に記憶していた更新結果情報を消去し処理を終了する。

【0206】

オペレーションパネル39の表示内容を、図34を用いて説明する。図34には、各種キー221、223、224、225、226と、表示画面222が示されている。表示画面222には、更新結果情報が有するプログラム名228と、更新前のバージョン229と更新後のバージョン230と、更新結果231と、更新日時232と、その他の定型文233と、OKボタン227が表示される。

【0207】

各種キー221、223、224、225、226は、融合機1の操作をするためのキーである。表示画面222は、オペレータに向け情報を表示するものである。OKボタン227は、表示されている内容を閉じる際にオペレータが押下するものである。オペレーションパネルに表示される内容も、プリントされる内容と同様のものとなる。

10

20

30

40

50

【0208】

次に、オペレータが閲覧できるように、NV-RAM47に更新履歴を記憶する処理について、図35のフローチャートを用いて説明する。なお、図35のフローチャートのステップS1301からステップS1308までは、図27のフローチャートのステップS1001からステップS1008と同じ処理のため、説明を省略し、ステップS1309以降の処理について説明する。

【0209】

リポート実行完了後、SCS20は、ステップS1309で、NV-RAM47に更新されたプログラムの情報があるかどうか判断する。

【0210】

NV-RAM47に更新されたプログラムの情報がない場合、SCS20は、ステップS1310でオペレーションパネル39に通常の画面を表示し、処理を終了する。

10

【0211】

NV-RAM47に更新されたプログラムの情報がある場合、SCS20は、ステップS1311で、記憶した更新結果情報を更新履歴情報に追記する。更新履歴情報に更新結果情報を追記した後、SCS20は、ステップS1312で、NV-RAM47に記憶していた更新結果情報を消去し処理を終了する。

【0212】

更新履歴情報を、図36を用いて説明する。更新履歴情報は、図36に示されるように、更新日時ごとに更新結果情報が有するプログラム名と、更新前のバージョンと更新後のバージョンと、更新結果と、更新日時とを有する。

20

【0213】

このように、更新するごとに、その結果が記憶される。例えば、図36の更新履歴情報には、2001年の12月15日午前4時0分に行われた更新から、2002年11月1日午前2時0分に行われた更新までの更新履歴が記憶されている。

【0214】

以上が本発明の実施例であるが、この実施例において、更新プログラムを格納する不揮発性メモリは、コストのかからないフラッシュメモリを用いるようにしても良い。

【図面の簡単な説明】

【0215】

【図1】本発明による融合機の一実施例の構成図である。

【図2】本発明による融合機の一実施例のハードウェア構成図である。

【図3】各処理部とプログラムの位置づけを示す図である。

【図4】各処理部の起動を示すシーケンス図である。

【図5】NV-RAMからRAMへの更新用プログラムの展開を示す図である。

【図6】通常の状態における通信の様子を示す図である。

【図7】更新用プログラム実行中の状態における通信の様子を示す図である。

【図8】優先情報を示す図である。

【図9】更新順番情報を生成する過程を示す図である。

【図10】更新順番情報を示す図である。

30

40

【図11】NV-RAMからRAMへの更新用プログラムの展開を示す図である。

【図12】更新順番情報の位置を示す図である。

【図13】プログラムの更新処理を示すフローチャートである。

【図14】直接にリポートを実行する処理を示すフローチャートである。

【図15】リポート要求をすることによりリポートを実行する処理を示すフローチャートである。

【図16】全てのプログラムの更新後にリポートを実行する処理を示すシーケンス図である。

【図17】更新管理テーブルを示す図である。

【図18】モジュールIDの構成を示す図である。

50

【図 19】更新管理テーブルを示す図である。

【図 20】プログラムの更新完了の順にリポートを実行する処理を示すフローチャートである。

【図 21】プログラムの更新完了の順にリポートを実行する処理を示すシーケンス図である。

【図 22】通信回線切断後にリポートを実行する処理を示すシーケンス図である。

【図 23】全てのプログラムの更新後にリポートを実行する場合の所要時間を示す図である。

【図 24】プログラムの更新順にリポートを実行する場合の所要時間を示す図である。

【図 25】更新進捗状況の表示に関する処理を示すフローチャートである。

10

【図 26】融合機の状態を保持する処理を示すフローチャートである。

【図 27】更新結果を印刷する処理を示すフローチャートである。

【図 28】更新結果情報を示す図である。

【図 29】更新結果情報を示す図である。

【図 30】印刷された更新結果を示す図である。

【図 31】更新結果を電子メールで送信する処理を示すフローチャートである。

【図 32】電子メールで送信された更新結果を示す図である。

【図 33】更新結果を表示する処理を示すフローチャートである。

【図 34】表示された更新結果を示す図である。

【図 35】更新履歴情報を記憶する処理を示すフローチャートである。

20

【図 36】更新履歴情報を示す図である。

【図 37】更新プログラムのデータ構造を示す図である。

【符号の説明】

【0216】

1 融合機

2 ソフトウェア群

3 融合機起動部

4 ハードウェア資源

5 アプリケーション層

6 コントローラ

30

7 コントロールサービス層

8 ハンドラ層

9 プリンタアプリ

10 コピーアプリ

11 ファックスアプリ

12 スキャナアプリ

13 ネットワークコントロールサービス (NCS)

14 リモートサービス (RS)

15 オペレーションパネルコントロールサービス (OCS)

16 ファックスコントロールサービス (FCS)

40

17 エンジンコントロールサービス (ECS)

18 メモリコントロールサービス (MCS)

19 オンデマンドアップデートサービス (OUS)

20 システムコントロールサービス (SCS)

21 システムリソースマネージャ (SRM)

22 ファックスコントロールユニットハンドラ (FCUH)

23 イメージメモリハンドラ (IMH)

24 その他のハードウェアリソース

25 スキャナ

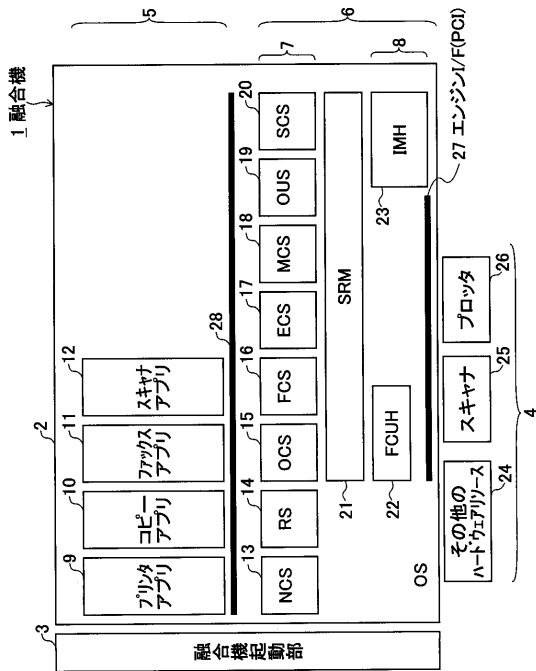
26 プロッタ

50

2 7	エンジン I / F (P C I)	
2 8	アプリケーションプログラムインターフェース (A P I)	
3 1	C P U	
3 2	システムメモリ (M E M - P)	
3 3	ノースブリッジ (N B)	
3 4	サウスブリッジ (S B)	
3 5	A G P (Accelerated Graphics Port)	
3 6	A S I C	
3 7	ローカルメモリ (M E M - C)	
3 8	ハードディスク装置 (H D)	10
3 9	オペレーションパネル	
4 0	ファックスコントロールユニット (F C U)	
4 1	U S B デバイス	
4 2	I E E E 1 3 9 4 デバイス	
4 3	エンジン部	
4 7	N V - R A M	
5 0	コントローラボード	
5 1	F C U ボード	
5 2	エンジンボード	
5 3	フィニッシャ	20
5 4	A D F	
5 5	操作パネルボード	
5 6	F C U ファーム	
5 7	プロッタファーム	
5 8	スキャナファーム	
5 9	フィニッシャファーム	
6 0	A D F ファーム	
6 1	操作パネルファーム	
6 2	P C I バス	
6 3	シリアル回線	30
6 4	同期シリアル回線	
7 0	優先度表	
7 1	更新順番情報	
7 2	システムソフト	
7 3	エンジンファーム	
1 0 1	N I C	
1 0 2	セントロニクスデバイス	
1 0 3	G 3 規格対応ユニット	
1 0 4	G 4 規格対応ユニット	
2 0 1、2 0 2	更新用プログラム	40
2 0 3	コントローラプログラム	
2 0 4	周辺ソフトプログラム	

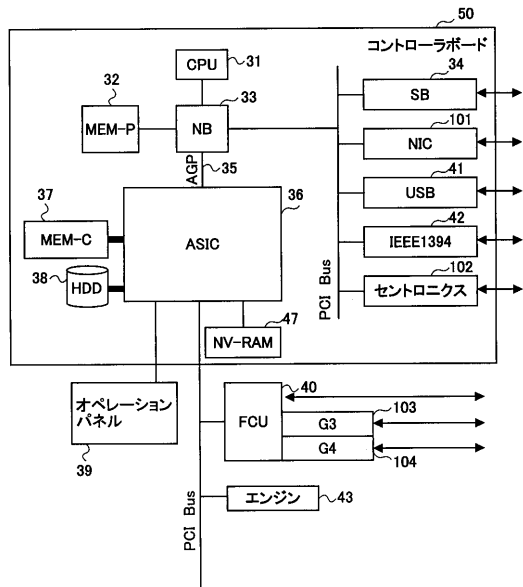
【図1】

本発明による融合機の一実施例の構成図



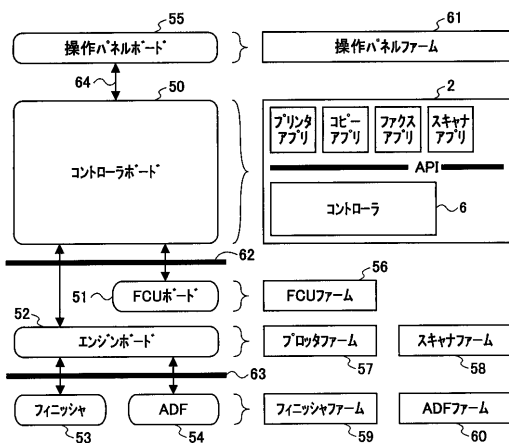
【図2】

本発明による融合機の一実施例のハードウェア構成図



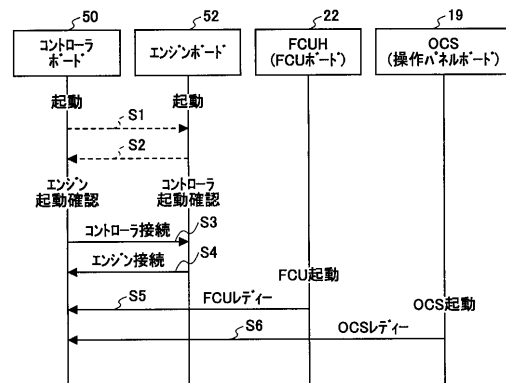
【図3】

各処理部とプログラムの位置づけを示す図



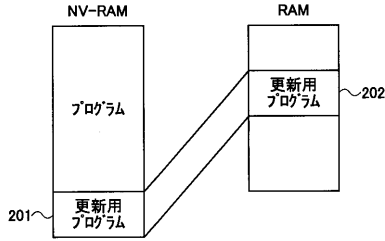
【図4】

各処理部の起動を示すシーケンス図



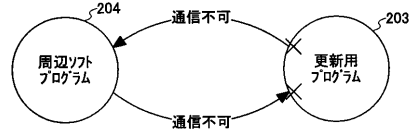
【図5】

NV-RAMからRAMへの更新用プログラムの展開を示す図



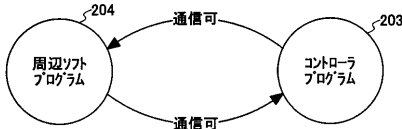
【図7】

更新用プログラム実行中の状態における通信の様子を示す図



【図6】

通常の状態における通信の様子を示す図



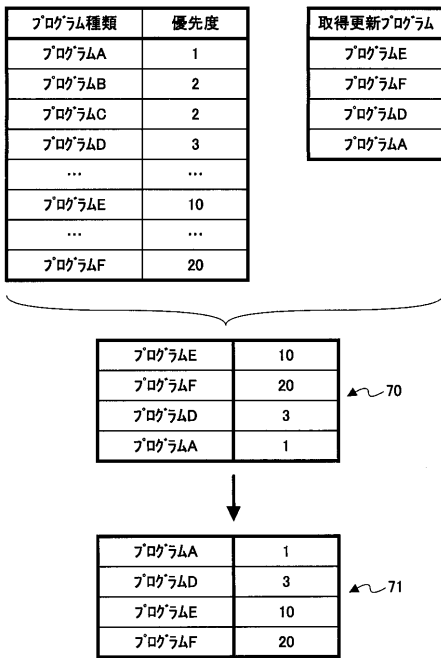
【図8】

優先情報を示す図

プログラム種類	優先度
プログラムA	1
プログラムB	2
プログラムC	2
プログラムD	3
...	...
プログラムE	10
...	...
プログラムF	20

【図9】

更新順番情報を生成する過程を示す図



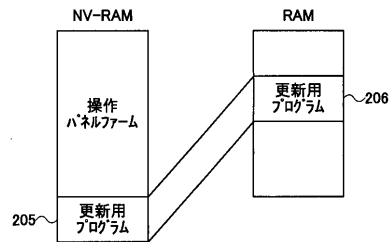
【図10】

更新順番情報を示す図

プログラム種類	バージョン	優先度	実行状態
フィニッシュファーム	Ver.2.11	1	未実行
FCUファーム	Ver.1.03	5	未実行
コントローラプログラム	Ver.1.22	10	未実行
操作パネルファーム	Ver.1.35	20	未実行

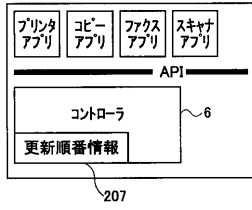
【図11】

NV-RAMからRAMへの更新用プログラムの展開を示す図



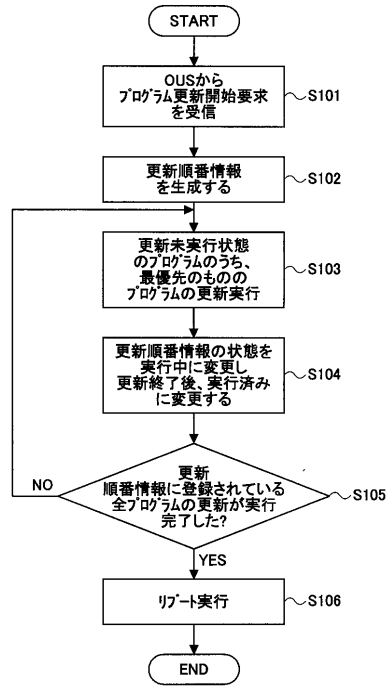
【図12】

更新順番情報の位置を示す図



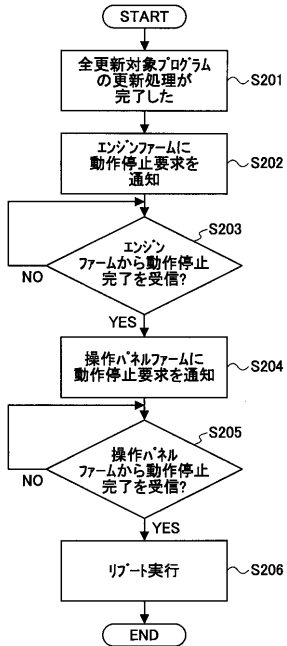
【図13】

プログラムの更新処理を示すフローチャート



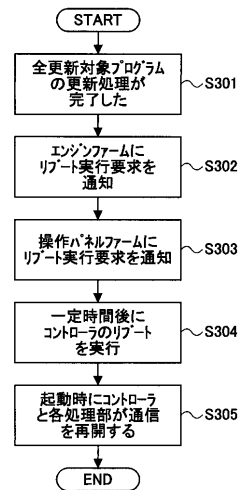
【図14】

直接にレポートを実行する処理を示すフローチャート



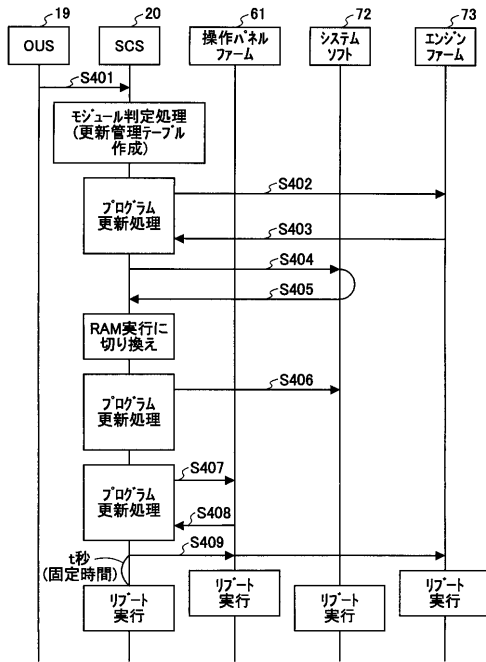
【図15】

レポート要求をすることによりレポートを実行する処理を示すフローチャート



【図16】

全てのプログラムの更新後にレポートを実行する処理を示すシーケンス図



【図17】

更新管理テーブルを示す図

プログラム種類	バージョン	モジュールID	実行状態
プロッタエンジンプログラム	Ver.1.03	JP2CFa_elpot	未実行
スキャナエンジンプログラム	Ver.1.22	JP2CFa_escan	未実行
コントローラプログラム	Ver.2.11	JP2CFd_system	未実行
プリンタアプリプログラム	Ver.1.35	JP2CFd_printer	未実行
FAXアプリプログラム	Ver.1.08	JP2CFa_fax	未実行
操作部プログラム	Ver.1.40	JP2CFe_anel	未実行

【図18】

モジュールIDの構成を示す図

プログラム種類	バージョン	モジュールID
プロッタエンジンプログラム	Ver.1.02	JP2CFa_elpot
スキャナエンジンプログラム	Ver.1.20	JP2CFa_escan
コントローラプログラム	Ver.2.07	JP2CFd_system
プリンタアプリプログラム	Ver.1.35	JP2CFd_printer
コピーアプリプログラム	Ver.1.05	JP2CFd_copy
操作部プログラム	Ver.1.39	JP2CFd_anel

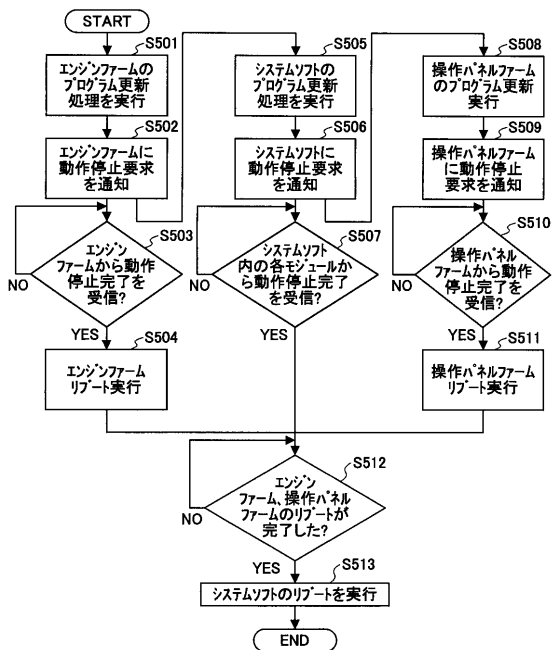
【図19】

更新管理テーブルを示す図

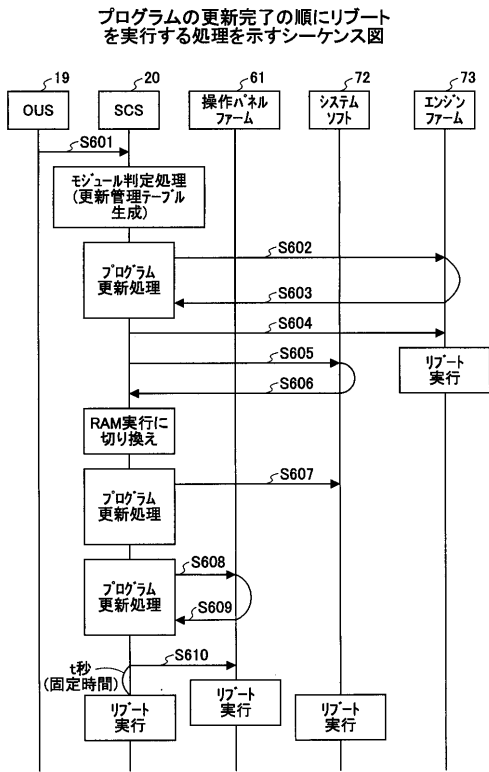
プログラム種類	バージョン	モジュールID	実行状態
プロッタエンジンプログラム	Ver.1.03	JP2CFa_elpot	未実行
スキャナエンジンプログラム	Ver.1.22	JP2CFa_escan	未実行
コントローラプログラム	Ver.2.11	JP2CFd_system	未実行
プリンタアプリプログラム	Ver.1.35	JP2CFd_printer	未実行
操作部プログラム	Ver.1.40	JP2CFd_anel	未実行

【図20】

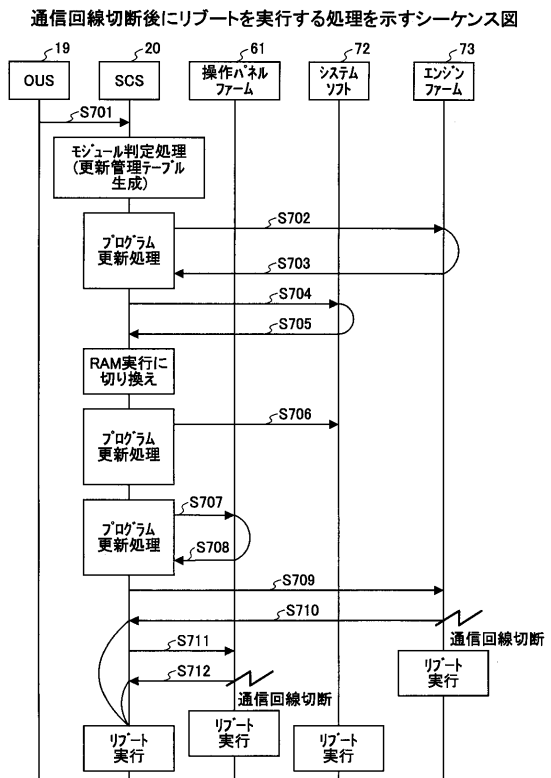
プログラムの更新完了の順にレポートを実行する処理を示すフローチャート



【図 2 1】

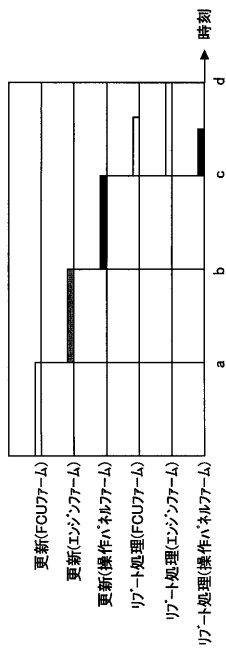


【図 2 2】



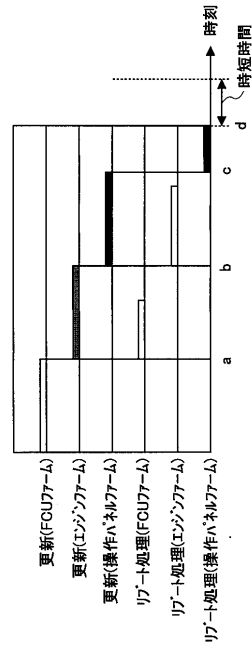
【図 2 3】

全てのプログラムの更新後にレポート
を実行する場合の所要時間を示す図



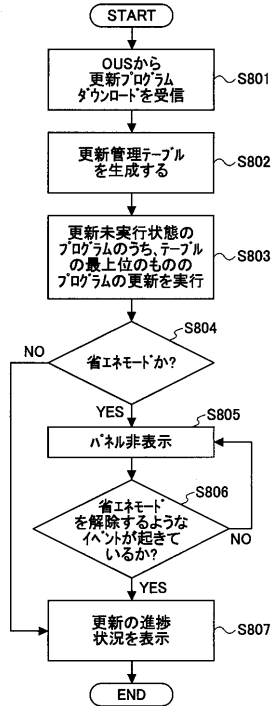
【図 2 4】

プログラムの更新順にレポートを実行する場合の所要時間を示す図



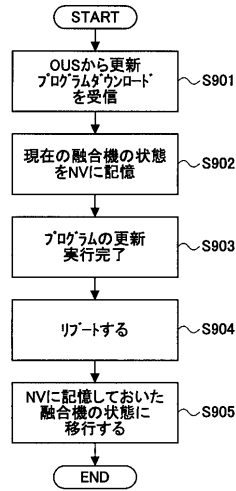
【図25】

更新進捗状況の表示に関する処理を示すフローチャート



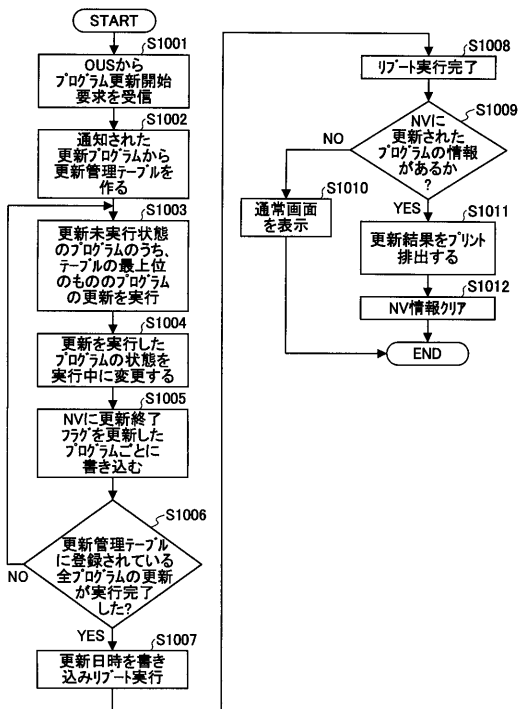
【図26】

融合機の状態を保持する処理を示すフローチャート



【図27】

更新結果を印刷する処理を示すフローチャート



【図28】

更新結果情報を示す図

プログラム名	更新前	更新後	更新結果
エンジンプログラム	V1.00	V1.03	正常
コントローラプログラム	V1.00	V1.03	正常
プリンタアプリプログラム	V1.00	NULL	NULL
操作部プログラム	V1.00	V1.01	正常

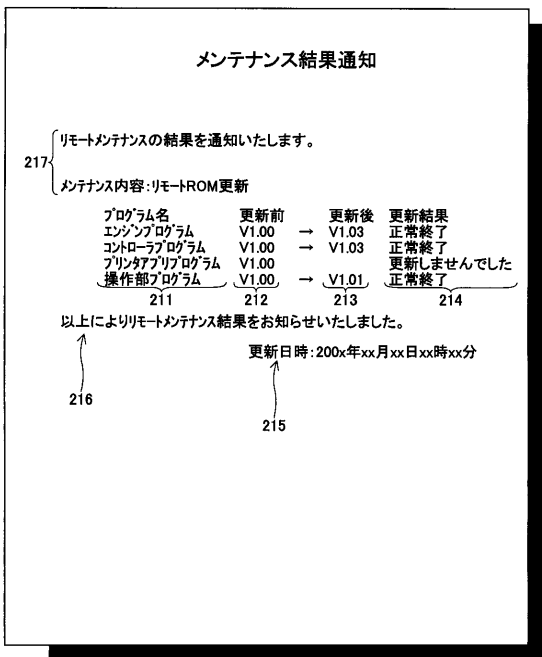
【図29】

更新結果情報を示す図

更新日時
yyyy.mm.dd.hh.mm

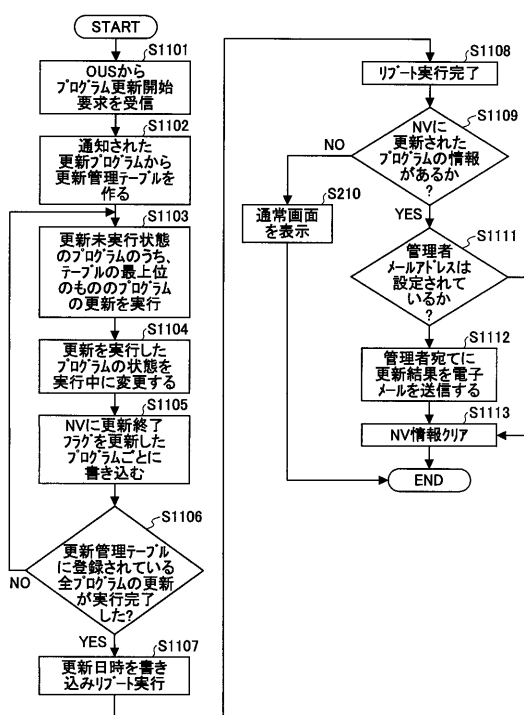
【図30】

印刷された更新結果を示す図



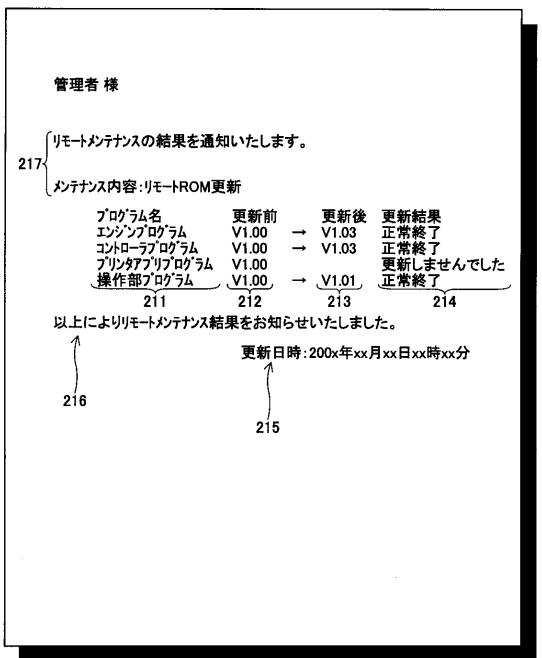
【図31】

更新結果を電子メールで送信する処理を示すフローチャート



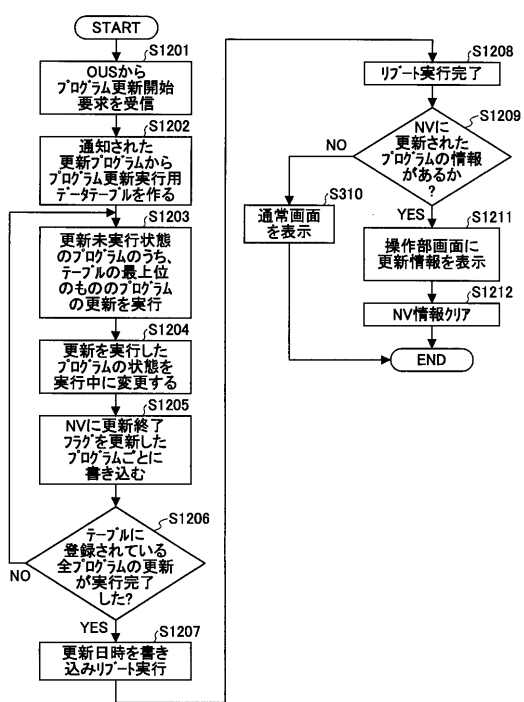
【図32】

電子メールで送信された更新結果を示す図



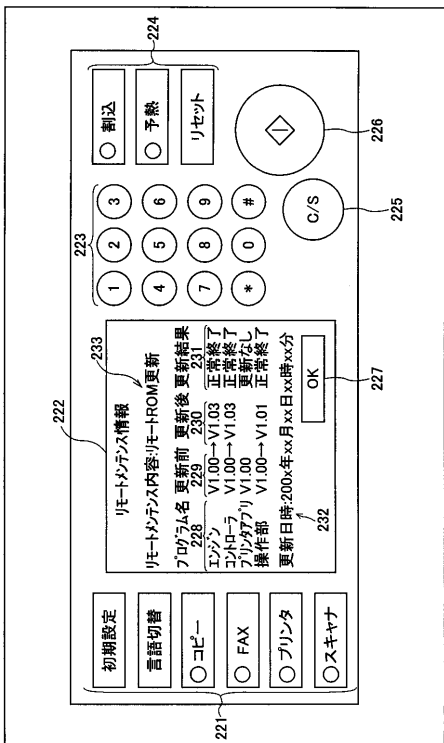
【図33】

更新結果を表示する処理を示すフローチャート



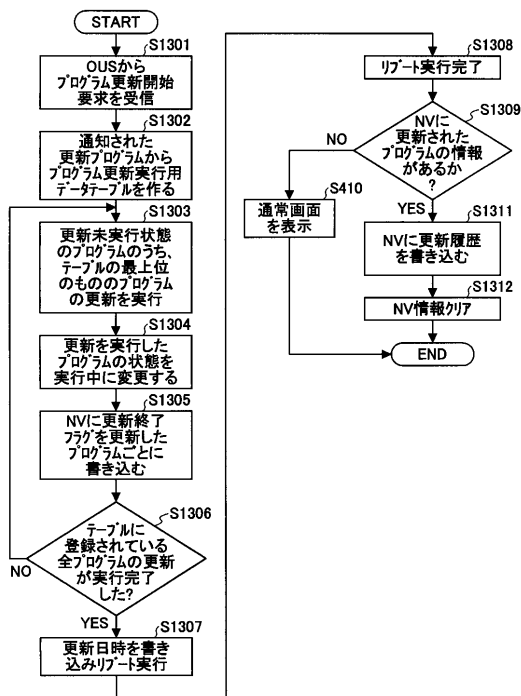
【図34】

表示された更新結果を示す図



【図35】

更新履歴情報を記憶する処理を示すフローチャート



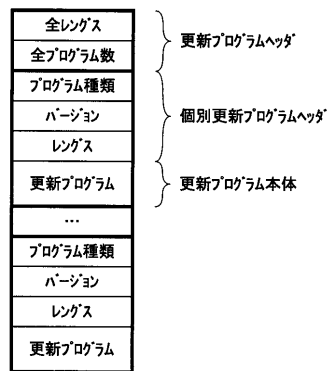
【図36】

更新履歴情報を示す図

プログラム名	更新前	更新後	更新結果	更新日時
エンジンプログラム	V1.00	V1.03	正常	2002.11.01.02.00
コントローラプログラム	V1.00	V1.03	正常	
プリンタプログラム	V1.00	NULL	NULL	
操作部プログラム	V1.00	V1.01	正常	
エンジンプログラム	V1.00	NULL	NULL	2002.08.05.02.00
コントローラプログラム	V0.90	V1.00	正常	
プリンタプログラム	V0.80	V1.00	正常	
操作部プログラム	V0.75	V1.00	正常	
...
エンジンプログラム	V0.01	V0.20	正常	2001.12.15.04.00
コントローラプログラム	V0.10	V0.20	正常	
プリンタプログラム	V0.20	V0.10	正常	
操作部プログラム	V0.15	V0.30	正常	

【図37】

更新プログラムのデータ構造を示す図



フロントページの続き

(72)発明者 佐々木 勝彦
東京都大田区中馬込1丁目3番6号 株式会社リコー内

審査官 渡邊 聡

(56)参考文献 特開2002-108582(JP,A)
特開2001-358733(JP,A)
特開平10-190898(JP,A)
特開平11-154099(JP,A)
特開2000-134242(JP,A)

(58)調査した分野(Int.Cl., DB名)
H04N 1/00