(19)日本国特許庁(JP)

(12)特許公報(B1)

(11)特許番号
特許第7539593号
(P7539593)

(45)発行日　令和6年8月23日(2024.8.23)

(24)登録日　令和6年8月15日(2024.8.15)

(51)国際特許分類
Ｇ０６Ｆ　8/71 (2018.01)

ＦＩ
Ｇ０６Ｆ　8/71

請求項の数　3　（全18頁）

(21)出願番号　特願2023-576433(P2023-576433)
(86)(22)出願日　令和5年6月20日(2023.6.20)
(86)国際出願番号　PCT/JP2023/022710
審査請求日　令和5年12月12日(2023.12.12)
早期審査対象出願

(73)特許権者　000006013
三菱電機株式会社
東京都千代田区丸の内二丁目７番３号
(74)代理人　100095407
弁理士　木村 満
(74)代理人　100131152
弁理士　八島 耕司
(74)代理人　100147924
弁理士　美恵 英樹
(74)代理人　100148149
弁理士　渡邉 幸男
(74)代理人　100181618
弁理士　宮脇 良平
(74)代理人　100174388
弁理士　龍竹 史朗

最終頁に続く

(54)【発明の名称】　プログラム可能機器、バージョン管理方法及びプログラム

(57)【特許請求の範囲】

【請求項１】

第１マスタ領域と第１ユーザ領域とを含み、前記第１マスタ領域に第１データセットを保存する第１記憶手段と、

前記第１記憶手段の前記第１マスタ領域に保存された前記第１データセットをバージョン管理する第１バージョン管理手段と、

前記第１バージョン管理手段によりバージョン管理された前記第１データセットを開発装置に送信する第１送信手段と、

前記第１送信手段が送信した前記第１データセットに基づいて作成され、前記開発装置にて前記第１データセットと対応付けてバージョン管理された第２データセットを受信する第１受信手段と、

を備え、

前記第１バージョン管理手段は、前記第１受信手段が受信した前記第２データセットを、前記第１データセットと対応付けて前記第１記憶手段の前記第１ユーザ領域に保存し、前記第１データセットと対応付けてバージョン管理する、

プログラム可能機器。

【請求項２】

プログラム可能機器が、

第１マスタ領域と第１ユーザ領域とを含む第１記憶手段の前記第１マスタ領域に保存された第１データセットをバージョン管理し、

バージョン管理された前記第１データセットを開発装置に送信し、
送信した前記第１データセットに基づいて作成され、前記開発装置にて前記第１データセットと対応付けてバージョン管理された第２データセットを受信し、
受信した前記第２データセットを、前記第１データセットと対応付けて前記第１記憶手段の前記第１ユーザ領域に保存し、前記第１データセットと対応付けてバージョン管理する、
バージョン管理方法。
【請求項３】
コンピュータに、
第１マスタ領域と第１ユーザ領域とを含む第１記憶手段の前記第１マスタ領域に保存された第１データセットをバージョン管理させ、
バージョン管理された前記第１データセットを開発装置に送信させ、
送信させた前記第１データセットに基づいて作成され、前記開発装置にて前記第１データセットと対応付けてバージョン管理された第２データセットを受信させ、
受信させた前記第２データセットを、前記第１データセットと対応付けて前記第１記憶手段の前記第１ユーザ領域に保存させ、前記第１データセットと対応付けてバージョン管理させる、
プログラム。
【発明の詳細な説明】
【技術分野】
【０００１】
本開示は、プログラム可能機器、バージョン管理方法及びプログラムに関する。
【背景技術】
【０００２】
プログラマブルロジックコントローラ、モーションユニットなどのＦＡ（Factory Automation）用のプログラム可能機器は、ユーザの開発により作成されたプログラムを実行して被制御機器を制御する。開発における変化点管理、進捗管理等のために、プログラムと当該プログラムに関連するデータとをバージョン管理することについて需要がある。
【０００３】
上記の事情に関連するものとして、例えば非特許文献１には、オープンソースの分散バージョン管理システムであるGitを活用して、複数人による制御ソフトウェア開発を支援する技術が開示されている。
【先行技術文献】
【非特許文献】
【０００４】
【文献】岩村　慎太郎，”オープンソースGitを活用したFA統合開発環境における複数人並行開発の実現”，[online]，2019/03/25，OMRON TECHNICS，[令和５年３月１０日検索]，インターネット＜URL：https://www.omron.com/jp/ja/technology/omrontechnics/2019/20190315-iwamura.html＞
【発明の概要】
【発明が解決しようとする課題】
【０００５】
非特許文献１に記載の技術において、プログラム可能機器そのものは分散バージョン管理システムとは独立したものとなっている。そのため、分散バージョン管理システムにて管理されているソースコードとプログラム可能機器で実行されるプログラムとの整合性を人手により管理する必要があり、ユーザの負担が大きくなる。例えば、プログラム可能機器で実行されたプログラムが、分散バージョン管理システムにて管理されているソースコードに基づいて生成されたものではない可能性が考えられる。
【０００６】
本開示の目的は、上記の事情に鑑み、プログラム可能機器に関連するデータを容易に分

散バージョン管理できるプログラム可能機器等を提供することにある。
【課題を解決するための手段】
【０００７】
上記の目的を達成するため、本開示に係るプログラム可能機器は、
第１マスタ領域と第１ユーザ領域とを含み、前記第１マスタ領域に第１データセットを保存する第１記憶手段と、
前記第１記憶手段の前記第１マスタ領域に保存された前記第１データセットをバージョン管理する第１バージョン管理手段と、
前記第１バージョン管理手段によりバージョン管理された前記第１データセットを開発装置に送信する第１送信手段と、
前記第１送信手段が送信した前記第１データセットに基づいて作成され、前記開発装置にて前記第１データセットと対応付けてバージョン管理された第２データセットを受信する第１受信手段と、
を備え、
前記第１バージョン管理手段は、前記第１受信手段が受信した前記第２データセットを、前記第１データセットと対応付けて前記第１記憶手段の前記第１ユーザ領域に保存し、前記第１データセットと対応付けてバージョン管理する。
【発明の効果】
【０００８】
本開示によれば、プログラム可能機器に関連するデータを容易に分散バージョン管理できる。
【図面の簡単な説明】
【０００９】
【図１】本開示の実施の形態１に係るバージョン管理システムの全体構成を示す図
【図２】本開示の実施の形態１に係るバージョン管理システムによるバージョン管理の一例を示す図
【図３】本開示の実施の形態１に係るＰＬＣの機能的構成を示すブロック図
【図４】本開示の実施の形態１に係るＰＬＣ及び本開示の実施の形態２に係る開発装置のハードウェア構成の一例を示す図
【図５】本開示の実施の形態１に係るＰＬＣによるバージョン管理の動作の一例を示すフローチャート
【図６】本開示の実施の形態１の変形例に係るバージョン管理システムの全体構成を示す図
【図７】本開示の実施の形態２に係る開発装置の機能的構成を示すブロック図
【図８】本開示の実施の形態２に係るバージョン管理システムによる分散バージョン管理の一例を説明する図
【図９】本開示の実施の形態２に係るバージョン管理システムによる分散バージョン管理の一例を説明する図
【図１０】本開示の実施の形態２に係るバージョン管理システムによる分散バージョン管理の一例を説明する図
【図１１】本開示の実施の形態２に係るバージョン管理システムによる分散バージョン管理の一例を説明する図
【図１２】本開示の実施の形態２に係るバージョン管理システムによる分散バージョン管理の一例を説明する図
【図１３】本開示の実施の形態２の変形例に係るバージョン管理システムによる分散バージョン管理の一例を説明する図
【図１４】本開示の実施の形態２の変形例に係るバージョン管理システムによる分散バージョン管理の一例を説明する図
【発明を実施するための形態】
【００１０】
以下、図面を参照しながら、本開示の実施の形態に係るバージョン管理システムを説明

する。各図面においては、同一又は同等の部分に同一の符号を付す。
【００１１】
（実施の形態１）
図１を参照しながら、実施の形態１に係るバージョン管理システム１を説明する。バージョン管理システム１は、ＰＬＣ（Programmable Logic Controller：プログラマブルロジックコントローラ）１０と複数の開発装置２０とを備える。バージョン管理システム１は、本開示に係るバージョン管理システムの一例である。
【００１２】
バージョン管理システム１においては、複数のユーザがそれぞれの開発装置２０によりＰＬＣ１０で実行されるプログラムを開発する。ユーザは、開発装置２０により開発したプログラムと当該プログラムに関連するデータとをまとめたデータセットをＰＬＣ１０に送信する。当該プログラムに関連するデータは、例えばプログラムのもととなるソースコード、プログラムに必要な設定ファイル及びリソースファイル、プログラムの仕様を説明するドキュメントファイルなどを含む。
【００１３】
ユーザは、例えばプログラムの変更作業が完了するたびに、データセットをＰＬＣ１０に送信する。ＰＬＣ１０は、受信したデータセットをバージョン管理しつつ、データセットに含まれるプログラムを実行する。ユーザが変更作業完了のたびにデータセットをＰＬＣ１０に送信する場合、ＰＬＣ１０は変更作業完了ごとのデータセットをバージョン管理することとなる。
【００１４】
ＰＬＣ１０は、ユーザが開発したプログラムを実行し、図示しない被制御機器を制御する。また、ＰＬＣ１０は、プログラムの実行のほか、上述のとおりデータセットをバージョン管理する。ＰＬＣ１０の機能的構成については後述する。ＰＬＣ１０は、本開示に係るプログラム可能機器の一例である。
【００１５】
開発装置２０は、ＰＬＣ１０で実行されるプログラムの開発に使用される。開発装置２０は、例えばエンジニアリングツールのプログラムがインストールされたパーソナルコンピュータである。ユーザは、開発装置２０によりＰＬＣ１０で実行されるプログラムを含むデータセットを作成し、作成したデータセットをＰＬＣ１０に送信する。開発装置２０は、本開示に係る開発装置の一例である。
【００１６】
次に、バージョン管理システム１におけるバージョン管理を概略的に説明する。バージョン管理システム１においては、概ね以下の流れでプログラムの開発が行われる。
（１）ユーザが開発作業によりプログラムを含むデータセットを作成。
（２）ＰＬＣ１０がユーザを認証。
（３）ユーザがＰＬＣ１０にデータセットを送信。
（４）ＰＬＣ１０が、受信したデータセットをバージョン管理。
（５）ユーザが、ＰＬＣ１０に送信したデータセットに含まれるプログラムをＰＬＣ１０上で実行してデバッグ作業。
（６）必要に応じて修正作業をして（１）からの作業を繰り返す。
（７）一通り修正作業が完了したら、プログラムをＰＬＣ１０上で実行して複数のユーザが動作確認をする。
（８）動作確認の結果に応じて、ユーザは（１）からの作業を繰り返す。
【００１７】
上記（１）－（８）のうち、（７）は複数のユーザが共同で行う作業であるが、その他はユーザが個別に行う作業である。そのため、ユーザの個別作業に関するデータセットと、共同作業に関するデータセットとを分けて管理できるとよい。
【００１８】
ＰＬＣ１０は、データセットを、マスタ領域とユーザごとのユーザ領域とに分けて管理

する。マスタ領域は、どのユーザでもアクセス可能な領域であり、ユーザ領域は、それぞれのユーザのみがアクセス可能であり他のユーザはアクセスできない領域である。ユーザが作成したデータセットは、原則としてユーザ領域に保存されるが、全ユーザ間で共有すべきデータセットは、ユーザの操作に基づいてマスタ領域に保存される。ユーザの操作に基づいてデータセットがマスタ領域に保存されることを、以下では「マスタ領域にデータセットを適用する」という。全ユーザ間で共有すべきデータセットとして、例えばユーザによる変更作業が一通り完成して全ユーザ間で共通してテストをするべき実行ファイルを含むデータセット、製品として出荷してよいレベルに達した実行ファイルを含むデータセットなどが挙げられる。

【００１９】

具体例を、図２を参照しながら説明する。図２に示す丸印は、１つのデータセットを表す。図２に示す矢印は、あるデータセットを直近の親として子のデータセットが作成されていることを示す。あるデータセットに対して変更作業を行い別のデータセットが作成されたとき、変更前のデータセットが「直近の親」であり、変更後のデータセットが「子」である。なお、「最初のデータセット」であるデータセットｘ１は、例えばＰＬＣ１０が初めてデータセットを受信したときに、当該データセットをマスタ領域に保存することにより作成される。

【００２０】

まず、ユーザＡが、マスタ領域にあるデータセットｘ１をもとに変更作業を行い、変更したデータセットをＰＬＣ１０に送信する。以下、データセットをもとに変更作業を行い、変更したデータセットをＰＬＣ１０に送信することを「更新」という。ユーザＡによる更新により、マスタ領域のデータセットｘ１から分岐したデータセットｙ１が、マスタ領域ではなくユーザＡ領域に保存される。データセットｘ１は、データセットｙ１の直近の親となる。

【００２１】

ユーザＡがデータセットｙ１を更新することにより、データセットｙ１を直近の親とするデータセットｙ２がユーザ領域Ａに保存される。続けて、同様の更新作業により、データセットｙ２を直近の親とするデータセットｙ３がユーザ領域Ａに保存される。

【００２２】

その後、ユーザＡは、データセットｙ３をマスタ領域に適用する作業を行うことにより、データセットｙ３と同内容のデータセットｘ２が、データセットｘ１とデータセットｙ３とを直近の親としてマスタ領域に保存される。このデータセットｘ２は、データセットｙ１からｙ３までの間のユーザＡによる変更作業の内容が適用されたものとなる。データセットｙ３の大元の親はデータセットｘ１であり、かつこの時点でのマスタ領域の最新のデータセットはデータセットｘ１であるため、問題なくデータセットｙ３をマスタ領域に適用することができる。なお、ここでいう「大元の親」とは、データセットの親を１世代以上遡ってたどり着くデータセットのことをいう。図２に示す例では、データセットｘ１は、データセットｙ１、ｙ２、ｙ３のいずれに対しても「大元の親」となる。

【００２３】

次に、ユーザＢの作業について説明する。まず、ユーザＢも、ユーザＡと同様に、マスタ領域のデータセットｘ１を更新することにより、マスタ領域のデータセットｘ１から分岐しデータセットｘ１を直近の親とするデータセットｚ１が、ユーザＢ領域に保存される。ユーザＢは、続けて更新を２回行い、データセットｚ１を直近の親とするデータセットｚ２がユーザＢ領域に保存され、データセットｚ２を直近の親とするデータセットｚ３がユーザ領域Ｂに保存される。なお、ここまでのユーザＢの作業は、マスタ領域にデータセットｘ２が保存される前に行われているものとする。

【００２４】

その後、ユーザＢは、自身の変更作業の内容が適用されたデータセットｚ３をマスタ領域に適用しようとする。しかし、マスタ領域に保存されている最新のデータセットがデータセットｘ１でなくデータセットｘ２となっているため、データセットｚ３をマスタ領域

に適用しようとしても競合している。そのため、このままでは適用することができない。
【００２５】
そこでユーザＢは、ユーザＢ自身の最新の変更作業が適用されたデータセットｚ３と、マスタ領域のデータセットｘ２とをマージする作業を行い、この作業にて変更されたデータセットにて更新する。これにより、データセットｚ３を直近の親とするデータセットｚ４がユーザ領域Ｂに保存される。
【００２６】
そしてユーザＢは、データセットｚ４をマスタ領域に適用する作業を行うことにより、データセットｚ４と同内容のデータセットｘ３が、データセットｘ２とデータセットｚ４とを直近の親としてマスタ領域に保存される。このデータセットｘ３は、データセットｚ１からｚ３までの間のユーザＢによる変更作業の内容と、データセットｘ２の内容とがマージされたものとなる。
【００２７】
ＰＬＣ１０は、データセットをマスタ領域とユーザ領域とに分けて管理することにより、データセットをユーザ単位で管理することができる。これにより、ユーザはプログラムの開発の際に、マスタ領域のデータセットと自身に対応するユーザ領域のデータセットのみに着目すればよく、複数人が開発する環境においてバージョン管理の運用が容易となる。
【００２８】
次に、図３を参照しながら、ＰＬＣ１０の機能的構成を説明する。ＰＬＣ１０は、記憶部１００と通信部１０１とユーザ認証部１０２とバージョン管理部１０３とプログラム実行部１０４とを備える。
【００２９】
記憶部１００は、後述のバージョン管理部１０３によるバージョン管理に基づいて、データセットを保存する。記憶部１００は、マスタ領域Ｍとユーザごとに設けられた複数のユーザ領域Ｕとを含む。記憶部１００は、本開示に係る第１記憶手段の一例である。
【００３０】
通信部１０１は、開発装置２０と通信する。通信部１０１は、例えばネットワークインタフェースである。通信部１０１は、本開示に係る第１送信手段及び第１受信手段の一例である。
【００３１】
ユーザ認証部１０２は、通信部１０１による通信先となっている開発装置２０に対応するユーザを認証する。ユーザ認証部１０２は、例えばパスワード認証によりユーザを認証する。あるいは、ＰＬＣ１０の管理者が予め開発装置２０ごとに対応するユーザをＰＬＣ１０に登録し、ユーザ認証部１０２は開発装置２０を識別することによりユーザを自動的に認証してもよい。
【００３２】
バージョン管理部１０３は、通信部１０１が受信したデータセットをバージョン管理する。バージョン管理部１０３は、データセットをマスタ領域Ｍと各ユーザ領域Ｕとに分けて保存して管理する。バージョン管理部１０３は、マスタ領域Ｍに保存されたデータセットを直近の親とするデータセットを、更新を行ったユーザに対応するユーザ領域Ｕに保存する。そのため、保存されるデータセットは、マスタ領域Ｍのデータセットから「分岐」したものとなる。また、バージョン管理部１０３は、ユーザ領域Ｕに保存されたデータセットを直近の親とするデータセットは、原則としてユーザ領域Ｕに保存する。そのため、原則として、ユーザが送信したデータセットはユーザ領域Ｕに保存される。
【００３３】
ただし、ユーザが、自身のユーザ領域Ｕに保存されたデータセットをマスタ領域Ｍに適用する作業を行うときには、バージョン管理部１０３は、当該データセットをマスタ領域Ｍに保存する。なお、ユーザがマスタ領域Ｍに適用しようとするデータセットの大元の親が、マスタ領域Ｍに保存された最新のデータセットと異なる場合には、バージョン管理部１０３は、ユーザによる適用を拒否する。この場合、ユーザは、マスタ領域Ｍに保存され

た最新のデータセットと自身のデータセットとをマージする作業を行う必要がある。
【００３４】
バージョン管理部１０３は、データセットをマスタ領域Ｍあるいはユーザ領域Ｕに保存するとき、データセット本体のみでなくデータセットを個別に識別するメタデータと、保存対象となるデータセットの直近の親のデータセットを示すメタデータとを、データセット本体と対応付けて保存する。メタデータとして、例えばデータセット本体を１つのバイナリデータとして扱ったときの当該バイナリデータのハッシュ値を使用することができる。これにより、データセットの保存時に直近の親のデータセットと対応付けることができるので、データセット間の親子関係を管理することができ、全体としてデータセットのバージョン管理をすることができる。バージョン管理部１０３は、本開示に係る第１バージョン管理手段の一例である。
【００３５】
プログラム実行部１０４は、バージョン管理部１０３を介して記憶部１００に保存されたプログラムを読み出し、読み出したプログラムを実行する。プログラム実行部１０４は、例えば記憶部１００に保存された最新のデータセットに含まれるプログラムを読み出して実行する。あるいは、プログラム実行部１０４は、ユーザが指定するデータセットに含まれるプログラムを読み出して実行してもよい。例えば、作業工程の都合上、ソースコードが一時的にコンパイルできないものとなり最新のデータセットに実行ファイルが存在しない状態となることがある。このような場合には、ユーザが実行ファイルの存在するデータセットを指定する必要が生じる。
【００３６】
また、プログラム実行部１０４は、プログラムの実行に関するログを、実行したプログラムに対応するデータセットと対応付けて記憶部１００に保存する。プログラムの実行に関するログは、例えば実行日時を示す情報、プログラム実行時に発生したイベント及びエラーに関する情報、実行したプログラムに対応するデータセットを示す情報などを含む。例えばユーザは、このログを分析することにより、データセットごとにプログラムの品質を評価することができる。
【００３７】
次に、ＰＬＣ１０のハードウェア構成の一例について、図４を参照しながら説明する。ＰＬＣ１０はプログラマブルロジックコントローラであるため、コンピュータの一種である。
【００３８】
ＰＬＣ１０は、バス１０００を介して互いに接続された、プロセッサ１００１と、メモリ１００２と、インタフェース１００３と、二次記憶装置１００４と、を備える。
【００３９】
プロセッサ１００１は、例えばＣＰＵ（Central Processing Unit：中央演算装置）である。プロセッサ１００１が、二次記憶装置１００４に記憶された動作プログラムをメモリ１００２に読み込んで実行することにより、ＰＬＣ１０の各機能が実現される。なお、プロセッサ１００１は、動作プログラムを実行するプロセッサとは別に、記憶部１００に保存されたデータセットを読み込んで実行するプロセッサを含んでもよい。この場合、後者のプロセッサによりプログラム実行部１０４の機能が実現される。
【００４０】
メモリ１００２は、例えば、ＲＡＭ（Random Access Memory）により構成される主記憶装置である。メモリ１００２は、プロセッサ１００１が二次記憶装置１００４から読み込んだ動作プログラムを記憶する。また、メモリ１００２は、プロセッサ１００１が動作プログラムを実行する際のワーキングメモリとして機能する。
【００４１】
インタフェース１００３は、例えばシリアルポート、ＵＳＢ（Universal Serial Bus）ポート、ネットワークインタフェースなどのＩ／Ｏ（Input/Output）インタフェースである。インタフェース１００３により通信部１０１の機能が実現される。

【００４２】
二次記憶装置１００４は、例えば、フラッシュメモリ、ＨＤＤ（Hard Disk Drive）、ＳＳＤ（Solid State Drive）である。二次記憶装置１００４は、プロセッサ１００１が実行する動作プログラムを記憶する。二次記憶装置１００４により記憶部１００の機能が実現される。
【００４３】
次に、図５を参照しながら、ＰＬＣ１０によるバージョン管理の動作の一例を説明する。図５に示す動作は、例えばユーザが開発装置２０を操作して、開発装置２０がデータセットをＰＬＣ１０に送信するために通信を開始した時点で実行される。
【００４４】
ＰＬＣ１０のユーザ認証部１０２は、開発装置２０と通信し、開発装置２０のユーザを認証する（ステップＳ１０１）。この動作により、データセットを送信したユーザが特定される。
【００４５】
ユーザが認証されると、開発装置２０はＰＬＣ１０にデータセットを送信し、ＰＬＣ１０の通信部１０１は、送信されたデータセットを受信する（ステップＳ１０２）。
【００４６】
ＰＬＣ１０のバージョン管理部１０３は、ユーザがマスタ領域Ｍにデータセットを適用しようとしているか否かを判定する（ステップＳ１０３）。
【００４７】
ユーザがマスタ領域Ｍにデータセットを適用しようとしていないとき（ステップＳ１０３：Ｎｏ）、バージョン管理部１０３は、ステップＳ１０２で受信したデータセットを、ステップＳ１０１で認証したユーザに対応したユーザ領域Ｕに保存する（ステップＳ１０４）。この動作において、当該保存されるデータセットの直近の親がマスタ領域Ｍにあるデータセットであるときには、当該保存されるデータセットはマスタ領域Ｍのデータセットから「分岐」したものとなる。そしてＰＬＣ１０は、バージョン管理の動作を終了する。
【００４８】
ユーザがマスタ領域Ｍにデータセットを適用しようとしているとき（ステップＳ１０３：Ｙｅｓ）、バージョン管理部１０３は、適用しようとしているデータセットの大元の親のデータセットと、マスタ領域Ｍの最新のデータセットとが一致するかを判定する（ステップＳ１０５）。
【００４９】
適用しようとしているデータセットの大元の親のデータセットと、マスタ領域Ｍの最新のデータセットとが一致するとき（ステップＳ１０５：Ｙｅｓ）、バージョン管理部１０３は、適用しようとしているデータセットをマスタ領域Ｍに保存する（ステップＳ１０６）。そしてＰＬＣ１０は、バージョン管理の動作を終了する。
【００５０】
適用しようとしているデータセットの大元の親のデータセットと、マスタ領域Ｍの最新のデータセットとが一致しないとき（ステップＳ１０５：Ｎｏ）、バージョン管理部１０３は、マスタ領域Ｍへのデータセットの適用を拒否する（ステップＳ１０７）。そしてＰＬＣ１０は、バージョン管理の動作を終了する。
【００５１】
以上、実施の形態１に係るバージョン管理システム１を説明した。バージョン管理システム１のＰＬＣ１０は、マスタ領域Ｍとユーザごとのユーザ領域Ｕとに分けてデータセットを保存してバージョン管理する。それにより、ユーザはプログラムの開発の際に、マスタ領域Ｍのデータセットと自身に対応するユーザ領域Ｕのデータセットのみに着目すればよく、複数人が開発する環境において容易にバージョン管理できる。
【００５２】
（実施の形態１の変形例）
実施の形態１において、図６に示すようにＰＬＣ１０は外部システム３０と通信可能に

接続され、ＰＬＣ１０のバージョン管理部１０３が、記憶部１００に保存されたデータセットを予め定められたタイミングで外部システム３０に送信してバックアップをとるようにしてもよい。バックアップのタイミングは、データセットが新たに追加されるタイミングであってもよいし、定期的に行われるものであってもよい。

【００５３】

また、ＰＬＣ１０ではなく開発装置２０が外部システム３０と通信可能に接続され、開発装置２０がＰＬＣ１０の記憶部１００に保存されたデータセットを読み出して外部装置３０に送信することによりバックアップをとるものであってもよい。例えば、開発装置２０はインターネットに接続可能だがＰＬＣ１０はインターネットに接続可能でない場合に、この形態を採用することができる。

【００５４】

実施の形態１において、各ユーザは自身に対応するユーザ領域Ｕのデータセットにはアクセスできる一方で、他のユーザに対応するユーザ領域Ｕのデータセットにはアクセスできないものとした。ここで、単にアクセスできないものとするのみでなく、各ユーザ領域Ｕのデータセットを、各ユーザに対応する暗号化キーにより暗号化してもよい。

【００５５】

また、マスタ領域Ｍのデータセットを、ＰＬＣ１０に固有の暗号化キーにより暗号化してもよい。これにより、ＰＬＣ１０を所有していない者はマスタ領域Ｍのデータセットを復号できないため、マスタ領域Ｍに保存されたデータセットが何らかの理由により漏洩しても第三者がデータセットの内容を知ることを防ぐことができる。

【００５６】

実施の形態１において、ユーザは、マスタ領域Ｍの最新のデータセットと競合しない限り、任意のタイミングでユーザ領域Ｕのデータセットをマスタ領域Ｍに適用できる。一方、マスタ領域Ｍに全く適用しないままデータセットの更新が続くと、マスタ領域Ｍのデータセットの内容とユーザ領域Ｕのデータセットの内容との乖離が大きくなり、開発に支障がでるおそれがある。そのため、例えばバージョン管理部１０３は、ユーザ領域Ｕ内でデータセットの更新が一定回数以上続いた場合に、ユーザに警告を出してもよい。例えば、バージョン管理部１０３が通信部１０１を介して警告を示すデータを開発装置２０に送信し、開発装置２０が当該警告をユーザに報知する。

【００５７】

（実施の形態２）

以下、実施の形態２に係るバージョン管理システム１を説明する。実施の形態２に係るバージョン管理システム１の全体構成は、図１に示す実施の形態１のものと同様である。実施の形態２においては、開発装置２０もデータセットのバージョン管理をする点が実施の形態１と異なる。

【００５８】

これにより、ユーザは、変更のたびに毎回ＰＬＣ１０にデータセットを送信する必要がなくなる。例えばユーザは、データセットを複数回変更し、変更作業が一通り完了してから、複数回の変更に対応する複数のデータセットをまとめてＰＬＣ１０に送信してもよい。

【００５９】

例えば図２に示す例において、実施の形態１において、ユーザＡはデータセットｙ１、ｙ２、ｙ３をそれぞれ毎回ＰＬＣ１０に送信することによりデータセットを更新していた。一方、実施の形態２において、ユーザＡはデータセットｘ１を取得すればその後はＰＬＣ１０と通信することなくデータセットｙ１、ｙ２、ｙ３についての変更作業をすることができ、データセットｙ１、ｙ２、ｙ３をまとめてＰＬＣ１０に送信してデータセットｙ３をマスタ領域Ｍに適用することができる。

【００６０】

図７を参照しながら、実施の形態２に係る開発装置２０の機能的構成を説明する。実施の形態２に係る開発装置２０は、記憶部２００と通信部２０１とデータセット作成部２０２とバージョン管理部２０３とを備える。

【００６１】
記憶部２００は、後述のバージョン管理部２０３によるバージョン管理に基づいて、データセットを保存する。記憶部２００は、マスタ領域Ｍとユーザ領域Ｕとを含む。ただし、ＰＬＣ１０の記憶部１００と異なり、ユーザ領域Ｕは、開発装置２０のユーザに対応する１つの領域のみとなる。記憶部２００は、本開示に係る第２記憶手段の一例である。
【００６２】
通信部２０１は、ＰＬＣ１０と通信する。後述のバージョン管理部２０３の制御に基づいて、通信部２０１は、開発装置２０によりユーザが作成したデータセットをＰＬＣ１０に送信し、ＰＬＣ１０から、記憶部１００に保存されたデータセットを受信する。通信部２０１は、例えばネットワークインタフェースにより実現される。通信部２０１は、本開示に係る第２受信手段及び第２送信手段の一例である。
【００６３】
データセット作成部２０２は、図示しない入力装置を介したユーザの操作に基づいて、ＰＬＣ１０にて実行されるプログラムと、当該プログラムに関連するデータとを作成する。つまりデータセット作成部２０２は、ユーザの操作に基づいてデータセットを作成する。データセット作成部２０２は、例えばソースコード、設定ファイル、ドキュメントなどを記述するためのエディタの機能を有する。データセット作成部２０２は、本開示に係るデータセット作成手段の一例である。
【００６４】
バージョン管理部２０３は、データセット作成部２０２が作成したデータセットを記憶部２００のユーザ領域Ｕに保存してバージョン管理する。バージョン管理部２０３は、データセットを保存する際に、ＰＬＣ１０のバージョン管理部１０３と同様に、データセット本体のみでなくデータセットを個別に識別するメタデータと、保存対象となるデータセットの直近の親のデータセットを示すメタデータとを、データセット本体と対応付けて保存する。これにより、開発装置２０内で、ユーザが作成したデータセットがバージョン管理されることとなる。
【００６５】
バージョン管理部２０３は、記憶部２００のユーザ領域Ｕに保存されたデータセットのうち、ＰＬＣ１０にまだ送信されていない１以上のデータセットを、通信部２０１を介してＰＬＣ１０にまとめて送信する。これにより、開発装置２０にてユーザが行った変更作業の内容をＰＬＣ１０に同期できる。
【００６６】
バージョン管理部２０３は、通信部２０１を介して、ＰＬＣ１０の記憶部１００のマスタ領域Ｍに保存されたデータセットのうち記憶部２００のマスタ領域Ｍに保存されていない１以上のデータセットを取得して記憶部２００のマスタ領域Ｍに保存する。これにより、ＰＬＣ１０のマスタ領域Ｍの変更内容を開発装置２０に同期できる。バージョン管理部２０３は、本開示に係る第２バージョン管理手段の一例である。
【００６７】
ここで、実施の形態２に係るＰＬＣ１０について、実施の形態１と異なる点を説明する。上述のとおり、実施の形態２に係る開発装置２０は、複数のデータセットをＰＬＣ１０に送信しうる。また、上述のとおり、実施の形態２に係る開発装置２０は、ＰＬＣ１０からマスタ領域Ｍのデータセットを取得する。そのため、実施の形態２に係るバージョン管理部１０３もこれに対応する。実施の形態２に係るバージョン管理部１０３は、複数のデータセットを開発装置２０からまとめて受信し、受信したデータセットをまとめて記憶部１００のユーザ領域Ｕに保存する。実施の形態２に係るバージョン管理部１０３は、開発装置２０の要求に応じて、マスタ領域Ｍの１以上のデータセットを開発装置２０に送信する。
【００６８】
開発装置２０のバージョン管理部２０３の機能及びＰＬＣ１０のバージョン管理部１０３の機能により、ＰＬＣ１０と開発装置２０とで個別にデータセットをバージョン管理し

つつ、必要に応じてＰＬＣ１０と開発装置２０とで同期をとることができる。そのため、実施の形態２に係るバージョン管理システム１によれば、分散バージョン管理を実現できる。

【００６９】

以下、図面を参照しながら具体例を説明する。初めはＰＬＣ１０の記憶部１００にはデータセットｘ１のみが保存されて、開発装置２０の記憶部２００には何らデータセットが保存されていないものとする。

【００７０】

まず、ユーザの操作に基づいて、開発装置２０のバージョン管理部２０３は、ＰＬＣ１０から記憶部１００に保存されたデータセットｘ１を取得して記憶部２００のマスタ領域Ｍに保存する。これにより、図８に示すように、開発装置２０の記憶部２００のマスタ領域Ｍにもデータセットｘ１が保存される。

【００７１】

次に、開発装置２０のユーザがデータセットｘ１に対する変更作業を行う。これにより、開発装置２０のデータセット作成部２０２が新たなデータセットを作成し、図９に示すように、バージョン管理部２０３が当該新たなデータセットを記憶部２００のユーザ領域Ｕにデータセットｙ１として保存する。この時点ではＰＬＣ１０と開発装置２０とは通信していないため、ＰＬＣ１０の記憶部１００には特に変化がない。

【００７２】

その後、ユーザが２回変更作業を行い、図１０に示すように、データセットｙ２、ｙ３が記憶部２００のユーザ領域Ｕに保存される。この時点でも、ＰＬＣ１０の記憶部１００には特に変化がない。

【００７３】

次に、ユーザの操作に基づいて、バージョン管理部２０３が、記憶部２００のユーザ領域Ｕに保存されたデータセットｙ１、ｙ２、ｙ３をまとめてＰＬＣ１０に送信する。ＰＬＣ１０のバージョン管理部１０３は、図１１に示すように、受信したデータセットｙ１、ｙ２、ｙ３を、記憶部１００のユーザ領域Ｕに保存する。この時点で初めて、開発装置２０とＰＬＣ１０とで同期をとることができる。

【００７４】

そしてユーザは、データセットｙ３をＰＬＣ１０の記憶部１００のマスタ領域Ｍに適用する操作を行うことにより、データセットｙ３と同内容のデータセットｘ２が記憶部１００のマスタ領域Ｍに保存される。開発装置２０のバージョン管理部２０３はこの後、ＰＬＣ１０から記憶部１００のマスタ領域Ｍに保存されたデータセットｘ２を取得し、記憶部２００のマスタ領域Ｍに保存する。これにより、図１２に示すように、ＰＬＣ１０、開発装置２０のいずれも、データセットｘ１、ｘ２、ｙ１、ｙ２、ｙ３をバージョン管理できることとなる。

【００７５】

なお、図２に示すデータセットｚ３、ｘ２、ｚ４及びｘ３に例示されるように、ユーザ領域Ｕのデータセットをマスタ領域Ｍに適用しようとするときに競合が発生してそのままでは適用できない場合がある。その場合には、まず、開発装置２０のバージョン管理部２０３がＰＬＣ１０から記憶部１００のマスタ領域Ｍに保存されたデータセットを取得し、記憶部２００のマスタ領域Ｍに保存する。次にユーザが、記憶部２００のマスタ領域Ｍに保存されたデータセットに対してマージ作業を行い、データセット作成部２０２がデータセットを作成し、バージョン管理部２０３がデータセットを記憶部２００のユーザ領域Ｕに保存する。そしてバージョン管理部２０３は、記憶部２００のユーザ領域Ｕに保存された新たなデータセットをＰＬＣ１０に送信し、ＰＬＣ１０のバージョン管理部１０３が受信したデータセットを記憶部１００のユーザ領域Ｕに保存する。その後ユーザは、再度ユーザ領域Ｕのデータセットをマスタ領域Ｍに適用しようとすればよい。

【００７６】

ＰＬＣ１０のプログラム実行部１０４は、実施の形態１と同様に、記憶部１００に保存

されたデータセットに含まれるプログラムを実行する。そのため、ユーザは、バージョン管理の対象となっているソースコードと、ＰＬＣ１０で実行されるプログラムとの整合性を容易に管理することができる。

【００７７】

なお、実施の形態２に係る開発装置２０のハードウェア構成は、図４に示すものと概ね同様のものとなる。これは、実施の形態１において説明したとおり、開発装置２０は、例えばエンジニアリングツールのプログラムがインストールされたパーソナルコンピュータであるからである。

【００７８】

以上、実施の形態２に係るバージョン管理システム１を説明した。実施の形態２に係るバージョン管理システム１によれば、ＰＬＣ１０と開発装置２０とで個別にデータセットをバージョン管理しつつ、必要に応じてＰＬＣ１０と開発装置２０とで同期をとることができる。そのため、実施の形態２に係るバージョン管理システム１によれば、分散バージョン管理を実現できる。また、実施の形態１と同様に、ＰＬＣ１０は記憶部１００に保存されたデータセットに含まれるプログラムを実行するので、ユーザは、バージョン管理の対象となっているソースコードと、ＰＬＣ１０で実行されるプログラムとの整合性を容易に管理することができる。そのため、実施の形態２に係るバージョン管理システム１によれば、プログラム可能機器に関連するデータを容易に分散バージョン管理できる。

【００７９】

また、データセットをマスタ領域Ｍとユーザ領域Ｕとで分けて管理することにより、ユーザが作成したデータセットをマージしようとする際に、ユーザ領域Ｕのデータセットとマスタ領域Ｍのデータセットとを確認し、意志決定をした上でマージをすることとなる。そのため、データセットをマスタ領域Ｍとユーザ領域Ｕとで分けて管理することにより、マージの際にユーザに大きな負担を与えることなく自然と意志決定の機会が得られることとなり、意図せずにデータセットがマージされてしまう事態を回避することができる。例えば、デバッグが完了していないデータセットが意図せずマージされてしまうことを回避できる。

【００８０】

（実施の形態２の変形例）

実施の形態２は実施の形態１の変形であるため、実施の形態１と同様、ユーザは複数であるものとした。一方、実施の形態２において、ユーザが単数であったとしても、ユーザの開発装置２０とＰＬＣ１０とでデータセットを分散バージョン管理することは可能である。この場合、ＰＬＣ１０の記憶部１００はマスタ領域Ｍと１つのユーザ領域Ｕとを有することとなる。また、ＰＬＣ１０は、ユーザ認証部１０２を必ずしも備える必要がなくなる。

【００８１】

また、実施の形態２において、マスタ領域Ｍとユーザ領域Ｕとに分けず１つの領域にてデータセットを管理する形態も考えられる。この形態において、複数のユーザが各開発装置２０にてデータセットを作成し、作成したデータセットをＰＬＣ１０に送信して更新する際の例を以下に説明する。

【００８２】

図１３に示すように、ユーザＡ、ユーザＢが、ＰＬＣ１０の記憶部１００に保存されたデータセットｘ１を元に、各開発装置２０にて開発を行う場合を考える。ユーザＡはデータセットｙ１、ｙ２、ｙ３を作成し、ユーザＢはデータセットｚ１、ｚ２を作成する。次に、ユーザＡがユーザＢよりも先にデータセットの更新を行い、その後にユーザＢがデータセットの更新を行うとする。この場合、図１４に示すように、先にデータセットを更新したユーザＡのデータセットｙ１、ｙ２、ｙ３はそのままＰＬＣ１０の記憶部１００に保存される。データセットｘ１、ｙ１、ｙ２、ｙ３が保存されているラインはいわゆる「マスターライン」となる。そして、後にデータセットを更新したユーザＢのデータセットｚ１、ｚ２は、マスターラインから分岐する形でＰＬＣ１０の記憶部１００に保存される。

このように、「早い者勝ち」でデータセットの更新を適用し、最先で更新されなかったデータセットについてはマスターラインから分岐する形で保存することにより、マスタ領域Ｍとユーザ領域Ｕとに分けることなく、複数のユーザによる分散バージョン管理を実現できる。なお、ｙ３とｚ２とのマージについては実施の形態１、２と同様に可能である。

【００８３】

また、実施の形態２において、ユーザ領域Ｕに、マスタ領域Ｍに長期間マージされないデータセットが存在する場合、ユーザに警告メールを送信するようにしてもよい。警告メールは、例えばユーザ認証部１０２にてユーザを認証した際に、認証に利用した情報を元に当該ユーザにメールを送信してもよい。長期間マスタ領域Ｍにマージされないデータセットは、マスタ領域Ｍのデータセットと乖離が大きくなっている可能性が高いため、警告をすることが有用である。

【００８４】

（その他の変形例）

上述の各実施の形態では、ＰＬＣ１０がデータセットをバージョン管理した。一方、ＰＬＣ以外の他のプログラム可能機器も、ＰＬＣ１０の各機能部と同様の機能部を備えることによりバージョン管理が可能となる。例えばモーションユニットで実行されるプログラムの開発において、モーションユニットがＰＬＣ１０の各機能部と同様の機能部を備えることにより、モーションユニットで実行されるプログラムに関するデータセットをバージョン管理することができる。

【００８５】

図４に示すハードウェア構成においては、ＰＬＣ１０及び開発装置２０が二次記憶装置１００４を備えている。しかし、これに限らず、二次記憶装置１００４をＰＬＣ１０及び開発装置２０の外部に設け、インタフェース１００３を介してＰＬＣ１０及び開発装置２０と二次記憶装置１００４とが接続される形態としてもよい。この形態においては、ＵＳＢフラッシュドライブ、メモリカードなどのリムーバブルメディアも二次記憶装置１００４として使用可能である。

【００８６】

また、図４に示すハードウェア構成に代えて、ＡＳＩＣ（Application Specific Integrated Circuit：特定用途向け集積回路）、ＦＰＧＡ（Field Programmable Gate Array）などを用いた専用回路によりＰＬＣ１０及び開発装置２０を構成してもよい。また、図４に示すハードウェア構成において、ＰＬＣ１０及び開発装置２０の機能の一部を、例えばインタフェース１００３に接続された専用回路により実現してもよい。

【００８７】

ＰＬＣ１０及び開発装置２０で用いられるプログラムは、ＣＤ－ＲＯＭ（Compact Disc Read Only Memory）、ＤＶＤ（Digital Versatile Disc）、ＵＳＢフラッシュドライブ、メモリカード、ＨＤＤ等のコンピュータ読み取り可能な記録媒体に格納して配布することが可能である。そして、かかるプログラムを特定の又は汎用のコンピュータにインストールすることによって、当該コンピュータをＰＬＣ１０及び開発装置２０として機能させることが可能である。

【００８８】

また、上述のプログラムをインターネット上の他のサーバが有する記憶装置に格納しておき、当該サーバから上述のプログラムがダウンロードされるようにしてもよい。

【００８９】

本開示は、本開示の広義の精神と範囲を逸脱することなく、様々な実施の形態及び変形が可能とされるものである。また、上述した実施の形態は、本開示を説明するためのものであり、本開示の範囲を限定するものではない。つまり、本開示の範囲は、実施の形態ではなく、請求の範囲によって示される。そして、請求の範囲内及びそれと同等の開示の意義の範囲内で施される様々な変形が、本開示の範囲内とみなされる。

【符号の説明】

【００９０】

１　バージョン管理システム、１０　ＰＬＣ、２０　開発装置、３０　外部システム、１００　記憶部、１０１　通信部、１０２　ユーザ認証部、１０３　バージョン管理部、１０４　プログラム実行部、２００　記憶部、２０１　通信部、２０２　データセット作成部、２０３　バージョン管理部、１０００　バス、１００１　プロセッサ、１００２　メモリ、１００３　インタフェース、１００４　二次記憶装置、Ｍ　マスタ領域、Ｕ　ユーザ領域。

【要約】

ＰＬＣ（１０）は、第１データセットを保存する記憶部（１００）と、記憶部（１００）に保存された第１データセットをバージョン管理するバージョン管理部（１０３）と、バージョン管理部（１０３）によりバージョン管理された第１データセットを開発装置に送信し、送信した第１データセットに基づいて作成され、開発装置にて第１データセットと対応付けてバージョン管理された第２データセットを受信する通信部（１０１）と、を備え、バージョン管理部（１０３）は、通信部（１０１）が受信した第２データセットを、第１データセットと対応付けて記憶部（１００）に保存し、第１データセットと対応付けてバージョン管理する。

【図面】

【図１】

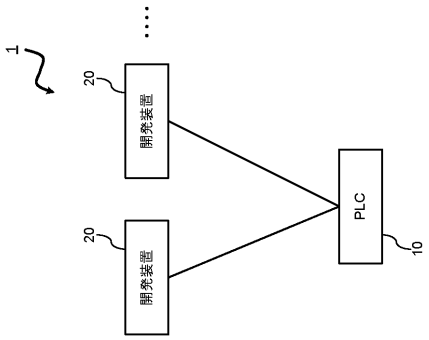


図1

【図２】

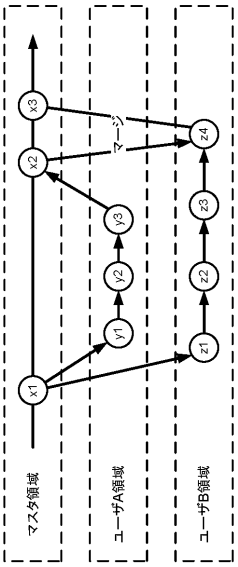


図2

【図３】

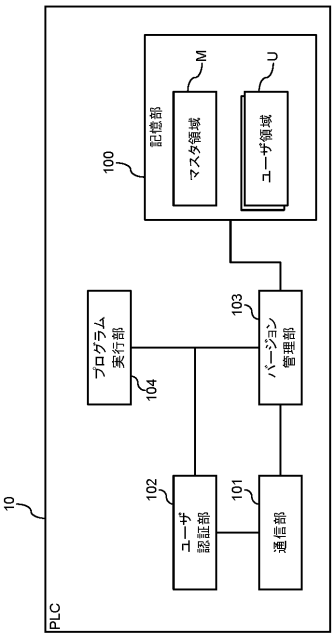


図3

【図４】

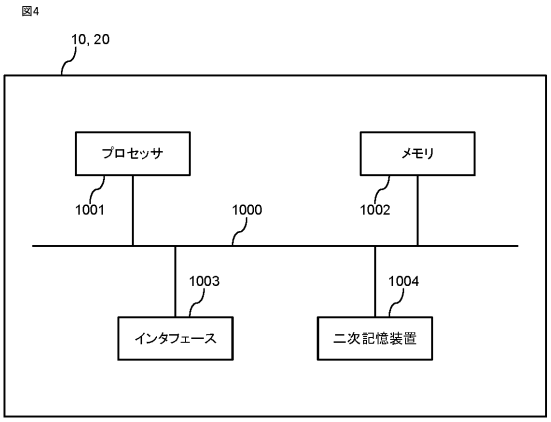


図4

【図５】

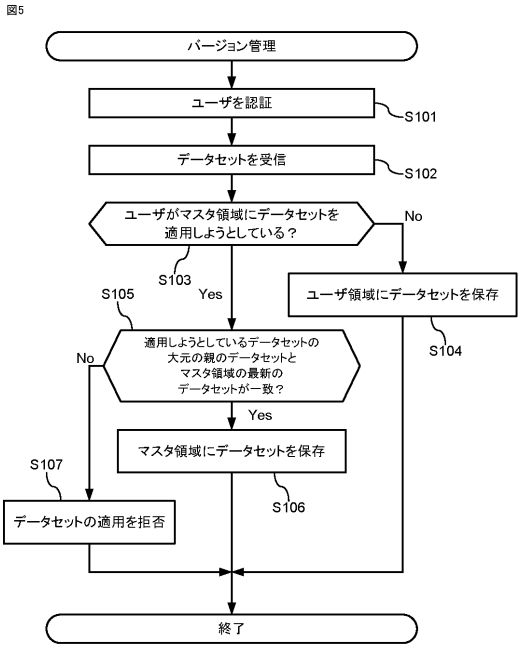


図5

【図６】

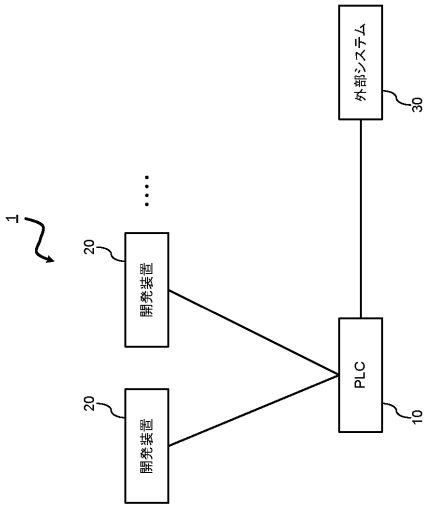


図6

【図７】

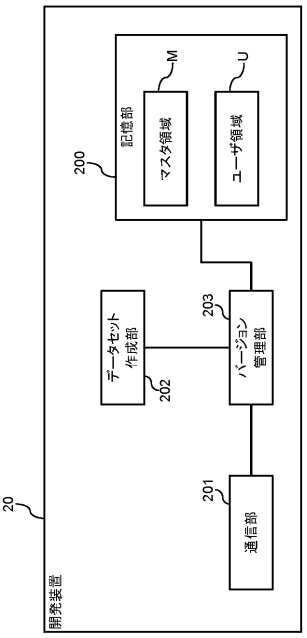


図7

【図８】

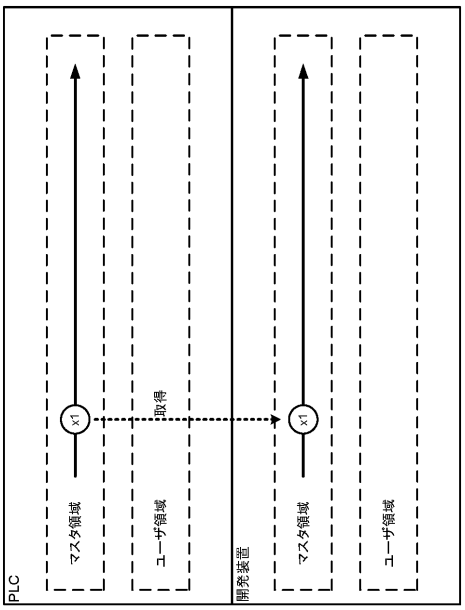


図8

【図９】

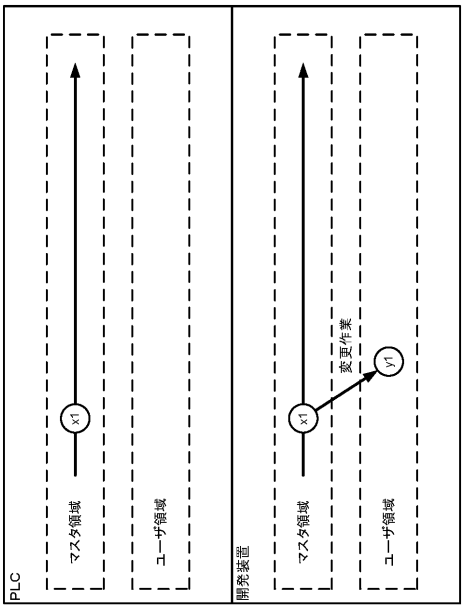


図9

【図１０】

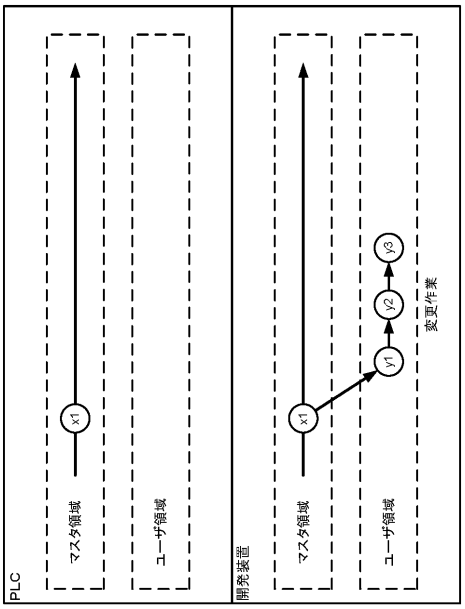


図10

【図１１】

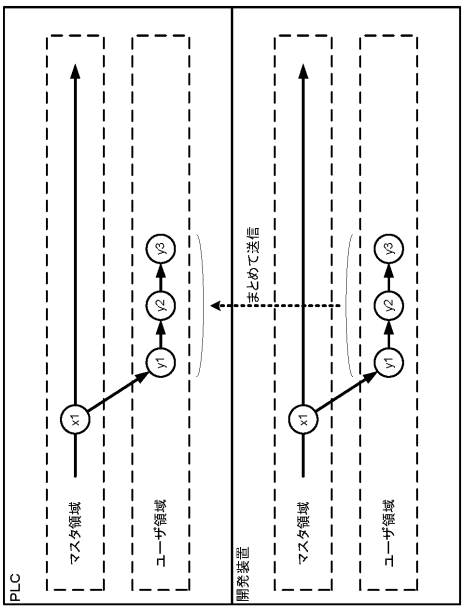


図11

【図１２】

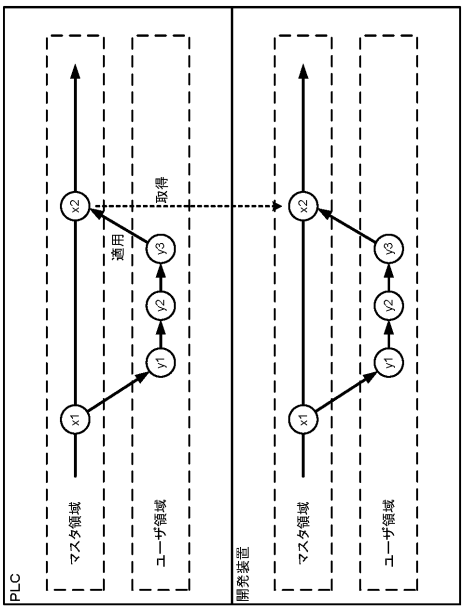


図12

【図１３】

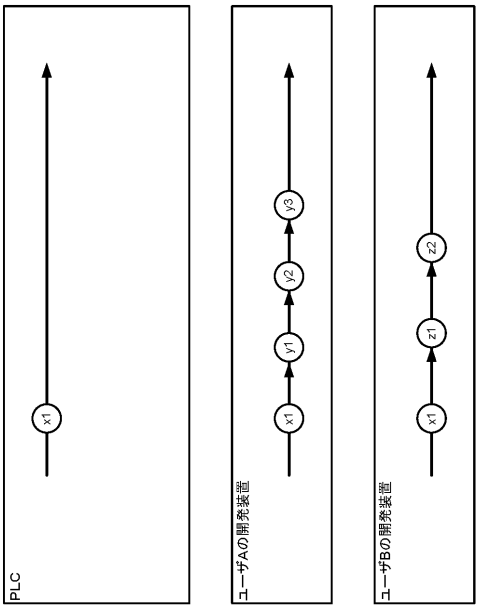


図13

【図１４】

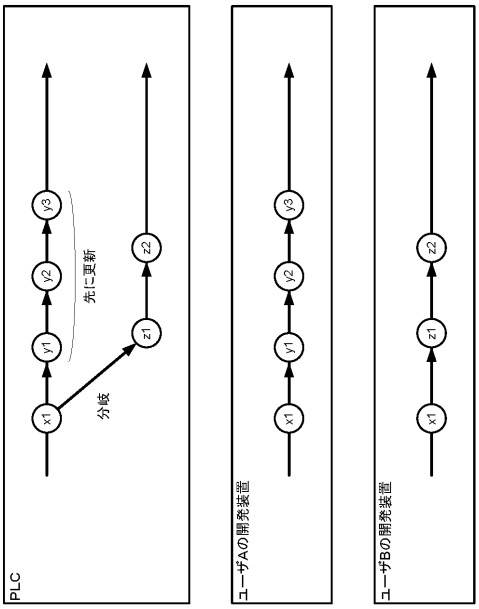


図14

フロントページの続き

(72)発明者　李 偉浩
東京都千代田区丸の内二丁目７番３号 三菱電機株式会社内
審査官　久々宇 篤志
(56)参考文献　特開平０６－３３２６７６（ＪＰ，Ａ）
特開２００８－２８２３６２（ＪＰ，Ａ）
特開２００１－３２５０９９（ＪＰ，Ａ）
特開平０８－３２０７８２（ＪＰ，Ａ）
特開平０７－０５６７２１（ＪＰ，Ａ）
特開２０２２－１３５４６４（ＪＰ，Ａ）
(58)調査した分野　(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　８／７１