

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4319105号
(P4319105)

(45) 発行日 平成21年8月26日 (2009. 8. 26)

(24) 登録日 平成21年6月5日 (2009. 6. 5)

(51) Int. Cl.	F I
H O 4 L 12/46 (2006. 01)	H O 4 L 12/46 Z
H O 4 Q 9/00 (2006. 01)	H O 4 Q 9/00 3 1 1 A

請求項の数 12 (全 27 頁)

(21) 出願番号	特願2004-219342 (P2004-219342)	(73) 特許権者	000006013
(22) 出願日	平成16年7月27日 (2004. 7. 27)		三菱電機株式会社
(65) 公開番号	特開2005-269597 (P2005-269597A)		東京都千代田区丸の内二丁目7番3号
(43) 公開日	平成17年9月29日 (2005. 9. 29)	(74) 代理人	100089118
審査請求日	平成18年12月15日 (2006. 12. 15)		弁理士 酒井 宏明
(31) 優先権主張番号	特願2004-41972 (P2004-41972)	(72) 発明者	中野 宣政
(32) 優先日	平成16年2月18日 (2004. 2. 18)		名古屋市東区矢田南五丁目1番14号 三
(33) 優先権主張国	日本国 (JP)		菱電機メカトロニクスソフトウェア株式会
			社内
		審査官	中木 努
		(56) 参考文献	特開2002-341906 (JP, A)
			)
			最終頁に続く

(54) 【発明の名称】 製造システム、ゲートウェイ装置、ゲートウェイプログラムおよび被制御装置の制御方法

(57) 【特許請求の範囲】

【請求項1】

制御装置と、該制御装置により制御される 1つ以上の 被制御装置と、前記制御装置と被制御装置との間で情報変換を行うゲートウェイ装置と、を備える製造システムであって、

前記制御装置は、記憶手段と第1通信手段とを備え、

前記記憶手段は、前記被制御装置に対する制御要求であって前記被制御装置の有する機能のみが抽象仕様で定義されて記述されるとともに制御対象となる前記被制御装置を選択する選択情報を含む制御要求情報を記憶し、

前記第1通信手段は、前記記憶手段に記憶された前記制御要求情報を前記ゲートウェイ装置に送信し、

前記ゲートウェイ装置は、変換手段とドライバ手段と第2通信手段とを備え、

前記変換手段は、前記制御装置から前記制御要求情報を受信し、該制御要求情報に記述された制御内容が前記選択情報により特定される制御対象となる前記被制御装置の有する機能である場合のみ、該制御要求情報を制御対象となる前記被制御装置の仕様に対応した前記ドライバ手段を制御するための、前記被制御装置の有する機能のみが抽象仕様で定義されたドライバ制御情報に該制御要求情報を変換して該ドライバ手段に送信し、

前記ドライバ手段は、異なる仕様の前記被制御装置に個別に対応するように複数設けられ、前記変換手段から受信した前記ドライバ制御情報を制御対象となる前記被制御装置の仕様に対応した被制御装置制御情報に変換し、

前記第2通信手段は、前記被制御装置制御情報を制御対象となる前記被制御装置に送信

10

20

し、

前記被制御装置は、前記ゲートウェイ装置から受信した前記被制御装置制御情報に従って前記被制御装置を制御する1つ以上の制御手段を備えること、
を特徴とする製造システム。

【請求項2】

前記ゲートウェイ装置の前記変換手段は、前記制御要求情報と前記ドライバ制御情報との対応情報を記憶した記憶手段を有し、該対応情報に基づいて前記制御要求情報を前記ドライバ制御情報に変換すること
を特徴とする請求項1に記載の製造システム。

【請求項3】

前記ゲートウェイ装置の前記ドライバ手段は、前記ドライバ制御情報と前記被制御装置制御情報との対応情報を記憶した記憶手段を有し、該対応情報に基づいて前記ドライバ制御情報を前記被制御装置制御情報に変換すること
を特徴とする請求項1に記載の製造システム。

【請求項4】

前記ゲートウェイ装置の前記ドライバ手段は、前記ドライバ制御情報に基づいて所定の機能を実行し完了すること
を特徴とする請求項1に記載の製造システム。

【請求項5】

被制御装置に対する制御要求であって前記被制御装置の有する機能のみが抽象仕様で定義されて記述されるとともに制御対象となる被制御装置を選択する選択情報を含む制御要求情報を送信する制御装置と、前記制御要求情報に基づいて被制御装置の制御を行う制御手段を備えて前記制御装置により制御される1つ以上の被制御装置と、の間で情報変換を行うゲートウェイ装置であって、

前記ゲートウェイ装置は、変換手段とドライバ手段と第2通信手段とを備え、

前記変換手段は、前記制御装置から前記制御要求情報を受信し、該制御要求情報に記述された制御内容が前記選択情報により特定される制御対象となる前記被制御装置の有する機能である場合のみ、制御対象となる前記被制御装置の仕様に対応した前記ドライバ手段を制御するための、前記被制御装置の有する機能のみが抽象仕様で定義されたドライバ制御情報に該制御要求情報を変換して該ドライバ手段に送信し、

前記ドライバ手段は、異なる仕様の前記被制御装置に個別に対応するように複数設けられ、前記変換手段から受信した前記ドライバ制御情報を制御対象となる前記被制御装置の仕様に対応した被制御装置制御情報に変換し、

前記第2通信手段は、前記被制御装置制御情報を制御対象となる前記被制御装置に送信すること、

を特徴とするゲートウェイ装置。

【請求項6】

前記変換手段は前記制御要求情報と前記ドライバ制御情報との対応情報を記憶した記憶手段を有し、該対応情報に基づいて前記制御要求情報を前記ドライバ制御情報に変換すること、

を特徴とする請求項5に記載のゲートウェイ装置。

【請求項7】

前記ドライバ手段は前記ドライバ制御情報と前記被制御装置制御情報との対応情報を記憶した記憶手段を有し、該対応情報に基づいて前記ドライバ制御情報を前記被制御装置制御情報に変換すること、

を特徴とする請求項5に記載のゲートウェイ装置。

【請求項8】

前記ドライバ手段は、前記ドライバ制御情報に基づいて所定の機能を実行し完了すること、

を特徴とする請求項5に記載のゲートウェイ装置。

10

20

30

40

50

【請求項 9】

被制御装置に対する制御要求であって前記被制御装置の有する機能のみが抽象仕様で定義されて記述されるとともに制御対象となる被制御装置を選択する選択情報を含む制御要求情報を送信する制御装置と、前記制御要求情報に基づいて被制御装置の制御を行う被制御手段を備えて前記制御装置により制御される 1 つ以上の被制御装置と、の間に情報変換を行うゲートウェイプログラムであって、

前記制御装置から受信した前記制御要求情報に記述された制御内容が前記選択情報により特定される制御対象となる前記被制御装置の有する機能である場合のみ、制御対象となる前記被制御装置の仕様に対応するとともに前記被制御装置の有する機能のみが抽象仕様で定義されたドライバ制御情報に前記制御要求情報を変換するステップと、

前記ドライバ制御情報を前記被制御装置の仕様に対応した被制御装置制御情報に変換するステップと、

をコンピュータに実行させることを特徴とするゲートウェイプログラム。

【請求項 10】

被制御装置に対する制御要求であって前記被制御装置の有する機能のみが抽象仕様で定義されて記述されるとともに制御対象となる被制御装置を選択する選択情報を含む制御要求情報に記述された制御内容が前記選択情報により特定される制御対象となる前記被制御装置の有する機能である場合のみ、前記被制御装置の仕様に対応するとともに前記被制御装置の有する機能のみが抽象仕様で定義されたドライバ制御情報に変換されるステップと、

前記ドライバ制御情報が、前記被制御装置の仕様に対応した被制御装置制御情報に変換されるステップと、

前記被制御装置が、前記被制御装置制御情報を受信し、当該被制御装置制御情報に従って制御されるステップと、

を含むことを特徴とする被制御装置の制御方法。

【請求項 11】

前記制御要求情報は、前記制御要求情報と前記ドライバ制御情報との対応情報を記憶した変換手段により、前記ドライバ制御情報に変換され、

前記ドライバ制御情報は、前記ドライバ制御情報と前記被制御装置制御情報との対応情報を記憶したドライバ手段により、前記被制御装置制御情報に変換されること、

を特徴とする請求項 10 に記載の被制御装置の制御方法。

【請求項 12】

前記ドライバ手段は、前記ドライバ制御情報に基づいて所定の機能を実行し完了すること、

を特徴とする請求項 11 に記載の被制御装置の制御方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、制御装置と、該制御装置により制御される被制御装置と、前記制御装置と被制御装置との間で情報変換を行うゲートウェイ装置とを備えた製造システムおよびこれを実現するゲートウェイ装置、ゲートウェイプログラム、被制御装置の制御方法に関するものである。

【背景技術】

【0002】

従来、生産設備として実際にワークの加工、洗浄、運搬などに携わる各種の工作機械や段取装置、洗浄装置、作業指示装置、搬送装置などの制御対象設備と、これらの制御対象設備の運用のための加工日程計画や加工順序情報、使用予定治具情報などの基本情報を処理して運転スケジュールを作成しながらこれらの制御対象設備を統括的に運転制御する制御装置とが、ネットワークで接続された構成を有するフレキシブル生産システムが提案され、普及してきている。

【0003】

このようなフレキシブル生産システムの一例としては、上記制御装置を上記制御対象設備の機能要素ごとに分割して複数配置して、機能要素ごとに制御対象設備の運転制御を行うようにすると共に、制御対象設備の運転制御に用いられる制御情報伝送用の通信回線と、フレキシブル生産システムの運用管理に用いられる運用情報伝送用の通信回線とを別々に設けて、システム全体としての情報の流れを円滑かつ高効率にしたものなどがある（たとえば、特許文献1参照）。

【0004】

【特許文献1】特開平1-234143号公報

【発明の開示】

【発明が解決しようとする課題】

【0005】

しかしながら、特許文献1に記載のフレキシブル生産システムなどにおいては、制御対象設備や制御装置は異なるベンダのものを組み合わせて構築される場合が多く存在する。このような場合、ベンダごとに制御対象設備に設定されるパラメータの意味や制御対象設備や制御装置で使用されるデータフォーマット、インターフェイスモデルなどが異なる。

【0006】

デファクト標準としてのコントローラ対応共通インターフェイスモデルを提供するものとして、たとえばEZSocket、MOII (Manufacturing Operation Information Interface)、ORIN (Open Robot Interface for The Network) などがあるが、これらは装置としての統一されたモデルではなく、装置を構成するコントローラとしての汎用的な機能をベースとしているため、装置の何をコントロールするかという具体的情報への変換をアプリケーションで実施する必要があった。

【0007】

また、これら以外にもデファクトとして各種のインターフェイスモデルが乱立されており、このようなインターフェイスモデルを使用する場合には、それぞれにそのアクセスを実行可能とするアプリケーションを設計する必要があった。すなわち、制御装置には、該制御装置が運転制御を行う制御対象設備ごとに（制御対象設備の仕様毎に）アプリケーションが必要となり、制御装置と制御対象設備との組み合わせに汎用性がないという問題があった。また、このアプリケーションを開発するための作業が必要であり、この作業を制御対象設備ごとに行うために、アプリケーションの開発者に大きな負荷が掛かるという問題もあった。

【0008】

本発明は、上記に鑑みてなされたものであって、ゲートウェイ装置による情報変換により制御装置と被制御装置の組み合わせの柔軟性に富んだ製造システム、およびこれを実現するゲートウェイ装置、ゲートウェイプログラム、被制御装置の制御方法を得ることを目的とする。

【課題を解決するための手段】

【0009】

上述した課題を解決し、目的を達成するために、本発明にかかる製造システムは、制御装置と、該制御装置により制御される1つ以上の被制御装置と、制御装置と被制御装置との間で情報変換を行うゲートウェイ装置と、を備える製造システムであって、制御装置は、記憶手段と第1通信手段とを備え、記憶手段は、被制御装置に対する制御要求であって被制御装置の有する機能のみが抽象仕様で定義されて記述されるときにも制御対象となる被制御装置を選択する選択情報を含む制御要求情報を記憶し、第1通信手段は、記憶手段に記憶された制御要求情報をゲートウェイ装置に送信し、ゲートウェイ装置は、変換手段とドライバ手段と第2通信手段とを備え、変換手段は、制御装置から制御要求情報を受信し、該制御要求情報に記述された制御内容が選択情報により特定される制御対象となる被制御装置の有する機能である場合のみ、該制御要求情報を制御対象となる被制御装置の仕

10

20

30

40

50

様に対応したドライバ手段を制御するための、被制御装置の有する機能のみが抽象仕様で定義されたドライバ制御情報に該制御要求情報を変換して該ドライバ手段に送信し、ドライバ手段は、異なる仕様の被制御装置に個別に対応するように複数設けられ、変換手段から受信したドライバ制御情報を制御対象となる被制御装置の仕様に対応した被制御装置制御情報に変換し、第2通信手段は、被制御装置制御情報を制御対象となる被制御装置に送信し、被制御装置は、ゲートウェイ装置から受信した被制御装置制御情報に従って被制御装置を制御する1つ以上の制御手段を備えること、を特徴とする。

【発明の効果】

【0010】

この発明によれば、被制御装置を制御するアプリケーションと被制御装置の組み合わせに自由度を持たせることができ、アプリケーションと被制御装置との種々の組み合わせに柔軟に対応可能な製造システムが実現できるという効果を奏する。

【発明を実施するための最良の形態】

【0011】

以下に、本発明にかかる製造システムおよびゲートウェイ装置、ゲートウェイプログラム、被制御装置の制御方法の実施の形態を図面に基づいて詳細に説明する。なお、本発明は以下の記述に限定されるものではなく、本発明の要旨を逸脱しない範囲において適宜変更可能である。

【0012】

まず、本発明の概要を従来の実情と比較して説明する。従来、生産設備として実際にワークの加工、洗浄、運搬などに携わる各種の工作機械などの制御対象設備と、これらの制御対象設備の運用のための加工日程計画や加工順序情報などの基本情報を処理して運転スケジュールを作成しながら、これらの制御対象設備を統括的に運転制御する制御装置とが、ネットワークで接続された構成を有するフレキシブル生産システムが提案されている。しかしながら、従来のフレキシブル生産システムでは、図1に示すように、制御装置のアプリケーションと該制御装置により制御される被制御装置とは1：1の関係にあり、各被制御装置に対しては該被制御装置に仕様毎に専用のアプリケーションが必要とされている。

【0013】

たとえば図1を参照して説明すると、被制御装置A20Aには該被制御装置A20Aの制御専用のアプリケーションA10Aが用意され、被制御装置B20Bには該被制御装置B20Bの制御専用のアプリケーションB10Bが用意される。そして、被制御装置M20Mには該被制御装置M20Mの制御専用のアプリケーションM10Mが用意され、被制御装置N20Nには該被制御装置N20Nの制御専用のアプリケーションN10Nが用意される。すなわち、被制御装置がN個ある場合には、それぞれを制御する専用のアプリケーションがN個必要とされる。これは、ベンダごとに被制御装置に設定されるパラメータの意味や被制御装置や制御装置で使用されるデータフォーマット、インターフェイスモデルなどが異なるためである。

【0014】

この場合、被制御装置A20Aを新製品の被制御装置AA（図示せず）に交換する際には、被制御装置AAをアプリケーションA10Aで制御することはできず、アプリケーションA10Aも被制御装置AA専用のアプリケーションAA（図示せず）に交換しなくてはならない。また、たとえばアプリケーションA10Aが故障した場合には、アプリケーションB10Bなど他のアプリケーションにより被制御装置A20Aを制御することはできず、アプリケーションA10Aと同じアプリケーションに入れ替えるか補修するまで被制御装置A20Aを動作させることができない。すなわち、従来の方式では、アプリケーションと被制御装置には組み合わせの自由度が無かった。同様に、1つの被制御装置内に仕様の異なるコントローラが複数存在する場合には、そのコントローラ毎にアプリケーションが必要となっていた。

【0015】

それに対して、本発明においては図2に示すようにゲートウェイ装置35を介して、制御装置31に備える一つのアプリケーション32（複数のアプリケーションでも可）で複数の被制御装置（被制御装置A40A～被制御装置N40N）を制御可能とするものである。これにより、たとえば被制御装置A40Aを被制御装置AA（図示せず）に交換した場合でも、既存のアプリケーション32により被制御装置AAを制御可能である。また、新たな被制御装置P（図示せず）を導入した場合においても、既存のアプリケーション32により被制御装置Pを制御することが可能である。

【0016】

すなわち、本発明は、アプリケーションと被制御装置との組み合わせに自由度をもたせることにより、柔軟性に富んだフレキシブル生産システムを実現するものである。なお、図2においては、アプリケーションを1つだけ表示してあるが、本発明においてアプリケーションは一つに限定されるものではなく、複数使用することも当然可能である。

【0017】

図3は、本発明にかかる製造システムの実施の形態の一例を示す概略構成図である。図3に示すように、本発明にかかる製造システムは、制御装置100と、ゲートウェイ装置110と、被制御装置140と、がそれぞれ通信回線154、156により接続されて構成される。

【0018】

制御装置100は、被制御装置140を制御するものであり、たとえば汎用の計算機などを使用することができる。制御装置100は、被制御装置140に対する制御要求が抽象仕様で記述された制御要求情報を記憶する記憶手段である記憶部104とゲートウェイ装置110に制御要求処理を実行するクライアントーエンティティ106とを備えて被制御装置140の制御を行うアプリケーション102と、ゲートウェイ装置110と通信する通信手段である通信部108と、を備えて構成される。

【0019】

アプリケーション102は、ユーザの所定の指示に基づいて被制御装置140を制御するために、ゲートウェイ装置110に制御要求情報を送信する。ユーザは、アプリケーション102が有する制御要求情報の中から所望の制御を選択することにより被制御装置140の制御を行うことができる。図3においては理解容易のためにアプリケーションを1つのみ記載してあるが、本発明においては、アプリケーションは単独ではなく制御装置内に複数存在することも可能である。

【0020】

ここで、被制御装置140に対する制御要求が抽象仕様で記述された制御要求情報は、制御装置100からアクセス（要求）可能な、すなわち被制御装置140が実行可能なインターフェイスの集合である。

【0021】

また、制御要求情報には後述するゲートウェイ装置110のドライバの選択情報も付随情報として含まれる。ドライバは後述するようにゲートウェイ装置内において複数存在可能であり、このドライバの選択情報は、ドライバが複数存在する場合に後述するゲートウェイ装置内の変換手段がドライバの選択を行う際に用いられる情報である。

【0022】

制御装置100からアクセス（要求）可能なインターフェイスは、被制御装置140およびその構成機器の設備仕様書、機械設計書、制御S/W設計書、制御盤設計書、コントローラ仕様書などから得られるものである。そして、制御装置100が複数の被制御装置140の制御要求情報、すなわちインターフェイスを有することにより、1つのアプリケーションで複数の被制御装置の制御を行うことが可能である。

【0023】

そして、このインターフェイスの内容は、被制御装置140のケーパビリティ記述における外部に提供するインターフェイスの全体集合、被制御装置140を構成する要素デバ

10

20

30

40

50

イス（構成機器）のケーパビリティ記述における外部に提供するインターフェイスの全体集合、後述するゲートウェイ装置１１０が提供するゲートウェイ装置独自のインターフェイス（ゲートウェイ装置独自ケーパビリティ）が該当する。ここで、ケーパビリティは、被制御装置１４０および該被制御装置を構成する各機器の機能をインターフェイス記述言語にて定義するものである。

【００２４】

このようなケーパビリティとしては、被制御装置１４０および該被制御装置を構成する各機器の実行する機能として、たとえばアクセス管理、ジョブ管理、運転管理、搬入／搬出管理、性能管理、診断、日時管理、イベント管理、ワーク管理、工具管理、レシピ／加工実行、アラーム管理、オブジェクト管理、レシピ／加工データ管理、などの種々の機能が記述される。また、このケーパビリティは、抽象仕様で記述される。すなわち、ケーパビリティ記述においては、各ベンダ特有のスペックなどの記述は除外されており、装置、機器の有する機能のみが定義され、抽象インターフェイス記述言語、たとえばIDL（Interface Defintion Language）により記述される。

10

【００２５】

通信部１０８は、ゲートウェイ装置１１０と通信する通信手段であり、各種の情報データの授受を行う。制御装置１００が被制御装置１４０の制御を行う際は、通信部１０８を介して制御要求情報がゲートウェイ装置１１０に送信され、また、被制御装置１４０およびゲートウェイ装置１１０から各種の情報が戻された場合には、通信部１０８においてそれらの情報を受信する。

20

【００２６】

ゲートウェイ装置１１０は、制御装置１００と被制御装置１４０との間にそれぞれと通信可能に接続され、制御装置１００と被制御装置１４０との環境の違いを吸収して、異なる仕様で構築された制御装置１００と被制御装置１４０との間の情報交換を可能とするものである。具体的には、制御装置１００と被制御装置１４０との間で情報データ変換を行う。これにより、制御装置１００からの情報は、ゲートウェイ装置１１０において、被制御装置１４０で解釈、動作可能なデータ情報に変換されて被制御装置１４０に入力される。また、被制御装置１４０からのデータ情報は制御装置１００で解釈可能なデータ情報に変換されて制御装置１００に入力される。これにより、制御装置１００と被制御装置１４０とが同一の仕様で構築されていなくても、その仕様の違いを吸収して制御装置１００と被制御装置１４０との間の情報交換を可能とし、汎用性の高い被制御装置１４０の制御が可能となる。

30

【００２７】

ゲートウェイ装置１１０は、第１の通信部１１２と、サーバーエンティティ１１４と、ドライバ１１６と、データベース１１８と、第２の通信手段１２０と、を備えて構成される。

【００２８】

第１の通信部１１２は、制御装置１００との通信を行う通信手段であり、各種の情報データの授受を行う。制御装置１００が被制御装置１４０の制御を行う際は、制御装置１００から送信された制御要求情報を通信部１１２が受信し、また、被制御装置１４０およびゲートウェイ装置１１０から各種の情報を制御装置１００に戻す場合には、通信部１１２においてそれらの情報を送信する。

40

【００２９】

サーバーエンティティ１１４は、上述したクライアントーエンティティ１０６とクライアントーサーバーアーキテクチャを構築し、クライアントーエンティティ１０６からの制御要求に応じて機能するものである。サーバーエンティティ１１４は、第１の通信部１１２において受信した制御装置１００（クライアントーエンティティ１０６）からの制御要求情報を、ドライバ１１６を動作させるためのドライバ制御情報に変換する変換手段である。ここで、制御要求情報のドライバ制御情報への変換は、サーバーエンティティ１１４が有する記憶部１２２に記憶された、制御要求情報とドライバ制御情報との対応情報リスト

50

に基づいて行われる。なお、ドライバ制御情報は未だ被制御装置の仕様に対応した形式ではなく、抽象記述形式の情報である。また、ドライバ116は後述するように複数存在可能であり、ドライバ116が複数存在する場合は、制御要求情報に含まれるドライバの選択情報に基づいてドライバの選択もこのサーバーエンティティ114において行われる。なお、制御要求情報とドライバ制御情報との対応情報リストは、データベース118に記憶する形態とすることも可能である。また、この対応情報リストには、後述するように被制御装置140からの返却値がサーバエンティティ114で解釈可能なデータ情報にドライバ116において変換された返却値情報を、制御装置100で解釈可能なデータ情報に変換するための逆変換対応情報も含まれている。なお、この逆変換対応情報は逆変換対応情報リストとして個別のリストとして保持しても良い。また、ここでの返却値とは、異常

10

【0030】

ドライバ116は、変換手段であるサーバーエンティティ114で変換された、ドライバ116を動作させるためのドライバ制御情報を、被制御装置140に送信する該被制御装置140およびその構成機器の仕様と、接続される通信環境と、に対応した被制御装置制御情報（各被制御装置140に対応した装置対応コマンド）に変換する。ドライバ116は、該ドライバ116の記憶部124に記憶されている、ドライバ制御情報と被制御装置制御情報（装置対応コマンド）との対応を記述した対応情報リストに基づいてドライバ制御情報を被制御装置制御情報（装置対応コマンド）に変換する。なお、ドライバ制御情報と被制御装置制御情報（装置対応コマンド）との対応を記述した対応情報リストは、データベース118に記憶する形態とすることも可能である。また、この対応情報リストには、被制御装置140で解釈可能なデータ情報として被制御装置140からゲートウェイ装置110に返却された返却値を、上位のサーバエンティティ114において解釈可能なデータ情報に変換するための逆変換対応情報リストも含まれている。なお、この逆変換対応情報は逆変換対応情報リストとして個別のリストとして保持しても良い。また、ここでの返却値とは、異常値情報、エラー値情報、動作確認情報など、下位のデバイスから送信されてくる種々のものを含む。

20

【0031】

このドライバ116は、被制御装置内において該被制御装置140をおよびその構成機器を制御するコントローラ144に1:1で対応するものであり、被制御装置140のコントローラ144と同数だけ用意される。たとえば、図4に示すように被制御装置が被制御装置164一つであり且つその中に含まれるコントローラがコントローラA166の一つである場合には、ゲートウェイ装置160にはコントローラ166に対応するドライバA162が一つ必要となる。また、図5に示すように被制御装置が被制御装置164の一つであり、その中に含まれるコントローラがコントローラA166とコントローラB170との二つである場合には、ゲートウェイ装置160にはコントローラA166およびコントローラB170にそれぞれ対応するドライバA162とドライバB168の二つが必要となる。また、図6に示すように被制御装置が被制御装置A164と被制御装置B172の二つであり、被制御装置A164にはコントローラA166が含まれ、被制御装置B172にはコントローラB170とコントローラC176とが含まれる場合には、ゲートウェイ装置A160にはコントローラA166、コントローラB170およびコントローラC176のそれぞれに対応するドライバA162とドライバB168とドライバC174との三つが必要となる。なお、図4～図6においては、ゲートウェイ装置および被制御装置において、ドライバおよびコントローラ以外の構成は省略して記載している。

30

40

【0032】

データベース118は、各種データファイルが記憶される記憶手段である。たとえばデータベース118が、制御要求情報とドライバ制御情報との対応を記述した対応情報リストファイルを記憶している場合は、サーバーエンティティでは、このデータベース118に記憶されたデータファイルに基づいて制御要求情報をドライバ制御情報へ変換する。ま

50

た、データベース 118 がドライバ制御情報と被制御装置制御情報（装置対応コマンド）との対応を記述した対応情報リストファイルを記憶している場合は、ドライバ 116 では、このデータベース 118 に記憶されたデータファイルに基づいてドライバ制御情報を被制御装置制御情報（装置対応コマンド）に変換する。

【0033】

また、データベース 118 には、リソース名変換ファイル 126 が記憶される。このリソース名変換ファイル 126 は、ケーパビリティ記述された被制御装置 140 およびその構成機器のインターフェイス内の特定ビットの実装アドレスと、その実装アドレスに対応する特定ビットの属性値名（論理名）との対応を記述した対応情報リストである。被制御装置 140 およびその構成機器においては、特定のハードウェアレジスタ（ワードレジスタ、バイトレジスタ）内の特定のビットを指定することが多い。この指定は、従来は実装アドレスで行われているが、実装アドレスは長い記述になるため、指定の間違いなどが生じやすく、またそのスペックがわかりづらい。

【0034】

そこで、本発明においては抽象インターフェイス言語（たとえば IDL）によるケーパビリティ記述において、上述した被制御装置 140 およびその構成機器の特定のビットを論理名で記述する。そして、ゲートウェイ装置 110 がデータベース 118 にリソース名変換ファイル 126 を有することにより、論理名をリソース名変換ファイル 126 に基づいて上述した特定のビットの実装アドレスに変換することができる。この変換はドライバ 116 がデータベース 118 内のリソース名変換ファイル 126 を参照することにより行われる。これにより、ドライバ 116 よりも上位の環境においては、被制御装置 140 およびその構成機器のリソースは、論理名により記述、特定され、情報伝達される。そして、ドライバ 116 よりも下位の環境、すなわち被制御装置 140 およびその構成機器内においては、従来と同様の実装アドレスで指定、情報伝達される。

【0035】

これにより、アプリケーション側から見た際に被制御装置 140 およびその構成機器の特定のビットのスペックを容易に認識することができ、また、指定の間違いなどを防止することができる。

【0036】

以下に、リソース名変換ファイル 126 に含まれる記述であり、任意のリソースの論理名－実装アドレスの対応記述の一例を示す。

【0037】

<リソース論理名－実装アドレスの対応記述（一例）>

「Xxx: abc1001. xyz2000. reg5. bit7」

【0038】

上記に示したリソースの論理名－実装アドレスの対応記述は、論理名「Xxx」は、「abc1001」というリソースの、「xyz2000」というコントローラの、「5番」のレジスタの、「16ビットの7番」、という実装アドレスに対応することを示している。

【0039】

第2の通信部 120 は、被制御装置 140 との通信を行う通信手段であり、各種の情報データの授受を行う。被制御装置 140 の制御を行う際は、第2の通信部 120 は、ドライバ 116 で変換された被制御装置制御情報（コマンド）を被制御装置 140 に送信する。また、被制御装置およびその構成機器から各種の情報を制御装置に戻す場合には、第2の通信部 120 においてそれらの情報を受信する。なお、ここでは、第1の通信部 112 と第2の通信部 120 とを独立して説明しているが、第1の通信部 112 と第2の通信部 120 との機能を一つの通信部で実行することも当然可能である。

【0040】

また、上記の制御装置 100 とゲートウェイ装置 110、およびゲートウェイ装置 110 と被制御装置 140 との接続はネットワークを介して接続されても良く、個別の通信回

10

20

30

40

50

線により接続されても良く、その接続形態は特に限定されず種々の接続形態を取ることが可能である。

【0041】

被制御装置140は、通信部142と、コントローラ144と、被制御装置140を構成するロボット146、CNC148、サーボ150などの構成機器（デバイス）を備えて構成される。

【0042】

通信部142は、ゲートウェイ装置110との通信を行う通信手段であり、各種の情報データの授受を行う。制御装置100が被制御装置140の制御を行う際は、ゲートウェイ装置110から送信された被制御装置制御情報（装置対応コマンド）を通信部142が受信し、また、被制御装置140およびその構成機器（デバイス）から各種の情報を制御装置100に戻す場合には、通信部142においてそれらの情報をゲートウェイ装置110に送信する。

【0043】

コントローラ144は、被制御装置制御情報に従って被制御装置140を制御する制御手段であり、通信部142において受信した、ゲートウェイ装置110のドライバ116から送信された被制御装置制御情報（装置対応コマンド）に従って、被制御装置140およびその構成機器を制御する。図3においてはコントローラを1つのみ記載しているが、このコントローラは被制御装置内においてデバイスの制御を分担させて複数設けることが可能である。このコントローラ144は、前述のゲートウェイ装置110のドライバ116と、コントローラ144の仕様別に、1：1で対応している。したがって、コントローラが複数存在する場合は、ゲートウェイ装置110にはそのコントローラの仕様に対応するドライバが設けられる。また、被制御装置140およびその構成機器（デバイス）から各種の情報を制御装置100に戻す場合には、該情報は、コントローラ144を介してゲートウェイ装置110に送信される。

【0044】

構成機器として、図3においてはロボット146、CNC148、サーボ150などの機器を記載しているが、本発明においては、被制御装置140の構成機器はこれらの機器に限定されるものでなく、種々の機器、およびさらに多くの機器を制御可能である。また、構成機器とコントローラとの接続はネットワークを介して接続されても良く、個別の通信回線により接続されても良く、その接続形態は特に限定されず種々の接続形態を取ることが可能である。

【0045】

つぎに、以上のように構成された製造システムにおける被制御装置140の制御について図7を用いて説明する。まず、ユーザが制御装置100において、制御装置からアクセス要求可能な被制御装置に対する要求事項（制御要求情報）を選択、入力し（ステップS11）、アプリケーション102を用いて制御要求情報をゲートウェイ装置110に送信する。この制御要求情報はクライアントエンティティ106から通信部108を介して送信される。

【0046】

ゲートウェイ装置110では、制御装置100から送信された制御要求情報を第1の通信部112で受信する。そして、サーバーエンティティ114がこの制御要求情報を、ドライバ116を動作させるためのドライバ制御情報に変換し、該ドライバ制御情報をドライバ116に送る（ステップS12）。ここで、制御要求情報のドライバ制御情報への変換は、サーバーエンティティ114が有する記憶部122に記憶された、制御要求情報とドライバ制御情報との対応情報リストに基づいて行われる。また、ドライバ116が複数存在する場合は、制御要求情報に含まれるドライバの選択情報に基づいてドライバの選択もこのサーバーエンティティ114において行われる。

【0047】

ドライバ116では、サーバーエンティティ114から送信されたドライバ制御情報を

、被制御装置 140 に送信する該被制御装置 140 およびその構成機器の仕様と、接続される通信環境と、に対応した被制御装置制御情報（装置対応コマンド）に変換する（ステップ 13）。すなわち、ドライバ 116 において、初めて制御指示が被制御装置 140 の仕様に対応した形式に変換される。ドライバ 116 は、該ドライバ 116 の記憶部 124 に記憶されている、ドライバ制御情報と被制御装置制御情報（装置対応コマンド）との対応を記述した対応情報リストに基づいてドライバ制御情報を被制御装置制御情報（装置対応コマンド）に変換する。そして変換した被制御装置制御情報（装置対応コマンド）を、第 2 の通信部 120 を介して被制御装置 140 に送信する。ここで、被制御装置 140 が複数ある場合、また、一つの被制御装置 140 内にコントローラ 144 が複数ある場合は、被制御装置制御情報（装置対応コマンド）内に含まれる被制御装置またはコントローラの選択情報により、被制御装置制御情報（装置対応コマンド）の送信先が適切に選択される。

10

【0048】

被制御装置 140 では、ゲートウェイ装置 110 のドライバ 116 からの被制御装置制御情報（装置対応コマンド）を通信部 142 で受信する。そして、コントローラ 144 が、この被制御装置制御情報（装置対応コマンド）に従って被制御装置 140 およびその構成機器を制御する（ステップ S14）。以上の流れにより、制御装置 140 により被制御装置 140 を制御することができる。また、被制御装置から制御装置へ送られる返却値が存在する場合は、上記と逆の処理が行われる。すなわち、被制御装置 140 からの返却値がゲートウェイ装置 110 に送られると、該返却値はサーバエンティティ 114 で解釈可能なデータ情報に変換するための逆変換がドライバ 116 において行われる。つぎに、変換された返却値はサーバエンティティ 114 に送られ、制御装置 100 で解釈可能な制御装置送信用の返却値情報に逆変換する。そして、変換された返却値情報が制御装置 100 に送信され、制御装置 100 ではこの返却値情報を受け取ることにより所定の処理を行う。

20

【0049】

上記のように、この製造システムにおいては、被制御装置 140 を制御するための制御要求情報が抽象記述されており、制御装置 140 とゲートウェイ装置 110 との間では、抽象記述された制御情報により制御指示の伝達がなされる。これにより、ゲートウェイ装置 110 は、抽象記述された制御要求情報を有する任意のアプリケーションと確実に制御指示の伝達を行うことが可能である。そして、アプリケーションが、抽象記述された制御要求情報を持つことで、被制御装置 140 の仕様に関わらず任意のアプリケーションを用いることが可能となり、アプリケーションの選択の自由度が非常に大きくなる。

30

【0050】

また、ゲートウェイ装置 110 が被制御装置 140 の仕様、詳しくはコントローラ 144 の仕様に対応したドライバ 116 を備えるため、制御装置 140 から抽象仕様で送られた制御指示情報を対応する被制御装置 140 の仕様に合わせた制御指示に変換することができる。これにより、被制御装置 140 を確実に制御することができる。そして、被制御装置を交換する場合には、新たな被制御装置に対応するドライバをゲートウェイ装置 110 に実装するだけで、アプリケーションを交換することなく、被制御装置を制御することができる。したがって、この製造システムにおいては、アプリケーションと被制御装置との組み合わせの自由度が非常に大きなものとされている。

40

【0051】

上記においては、制御の流れとして図 3 に示すようにドライバ 116 からコントローラ 144 に被制御装置制御情報（装置対応コマンド）が送られて被制御装置 140 またはその構成機器の制御が行われる場合について説明したが、本発明においては、コントローラ 144 を介さずに制御を行う形態とすることも可能である。たとえば、制御装置 140 からの制御要求内容が、デバイス内にあるリソースの特定のビットに関する情報の収集である場合を考える。この場合、サーバエンティティ 114 からドライバ制御情報が専用のデバイスドライバ 116 a に送られ、該デバイスドライバ 116 a から被制御装置 140

50

にデバイス制御情報（デバイス対応コマンド）が送られる。そして、デバイスドライバ 116 a からコントローラ 144 を介さずに図 3 の矢印 S のように被制御装置 140 内のネットワーク 152 にアクセスしてデバイス内、たとえば CNC 内にあるリソースの特定のビットに関する情報の収集を行うという形態も取ることができる。

【0052】

また、上記においては、制御の流れとして図 8 に示すようにサーバーエンティティ 114 からドライバ制御情報がドライバ 116 に送られ、ドライバ 116 から被制御装置 140（装置）に被制御装置制御情報（装置対応コマンド）が送られる形態について説明したが、本発明においては、この他の形態を取ることも可能である。たとえば、図 9 に示すようにサーバーエンティティ 114 からまず第 1 ドライバ 116 b にドライバ制御情報が送られ、第 1 ドライバ 116 b で所定のコード変換が行われたドライバ制御情報が第 2 ドライバ 116 c に送られ、該第 2 ドライバ 116 c から被制御装置 140（装置）に被制御装置制御情報（装置対応コマンド）が送られる形態を取ることもできる。また、図 10 に示すようにサーバーエンティティ 114 からドライバ制御情報がドライバ 116 d で処理を完了する形態も取ることができる。この場合、ドライバ 116 d 自体が所定の機能を実行する。実行する機能内容としては、たとえば、ドライバ 116、デバイスドライバ 116 a 対応コールの排他制御、インターフェイス実施履歴、インターフェイス実施のタイムスタンプと履歴のファイリング、インターフェイス授受データ構造の変換（コード、ビットバイトの変換など）などが一例として挙げられる。

【0053】

また、以上のようなゲートウェイ装置 110 は、つぎのようにして構築することができる。まず、被制御装置 140 および該被制御装置 140 のデバイスの設備仕様書、機械設計書、制御 S/W 設計書、制御盤設計書、コントローラ仕様書など設計情報や機能情報、すなわちケーパビリティを共通の記述モデルに従って記述する。具体的には、たとえば抽象インターフェイス記述言語である IDL により記述する。これらのケーパビリティは、被制御装置 140 およびデバイスの設計仕様書、機械設計書等からその全てを抽出して記述する。したがって、このケーパビリティには、被制御装置 140 およびその構成機器が実行可能な全ての機能が抽象インターフェイス記述言語により記述されており、このケーパビリティにより被制御装置 140 およびその構成機器の機能を把握することができる。

【0054】

このようにケーパビリティを共通の記述モデルに従って記述することにより、設計データの削除、新規設計データの追加、設計データの検索などを行う必要が生じた際にも、アプリケーションを開発した人間以外の第 3 者が、それらの作業を容易に行うことができる。その結果、開発者の不在、転勤、退職等によりアプリケーションの開発・メンテナンス作業環境の変化があった場合でも、任意の第 3 者がこれらの作業を容易に行うことが可能である。したがって、ケーパビリティを共通の記述モデルに従って記述することにより、アプリケーションの開発・メンテナンスの自由度が非常に大きなものとなる。

【0055】

抽象インターフェイス記述言語によりケーパビリティを記述する場合には、たとえば図 11 に示すようなテーブルの領域 200 にクラスを記述し、領域 210 に属性およびそのパラメータを記述し、領域 220 にオペレーション内容を記述する。そして、全ての被制御装置およびその構成機器の全ての機能について同様に記述を行い、ケーパビリティファイルを作成する。このようにして、装置ケーパビリティファイル（被制御装置のケーパビリティ）と、装置デバイスケーパビリティファイル（被制御装置の構成機器のケーパビリティ）とが作成される。また、図 12 に IDL による記述例を示す。なお、実際の形態においては、ここまでのケーパビリティファイルの作成は被制御装置のメーカーにおいて行われ、作成されたケーパビリティファイルがメーカーから提供される。

【0056】

また、被制御装置の個別のドライバを準備する。この個別のドライバも実際の形態においては、被制御装置のメーカーから提供される。ドライバはたとえば C 言語のソースコー

ド（バイナリファイル）などで提供されるものとする。

【0057】

その後、ケーパビリティをゲートウェイ装置に実装する。また、制御要求情報とドライバ制御情報との対応を記述した対応情報リストファイルをゲートウェイ装置110のサーバーエンティティ114に実装する。また、被制御装置140のコントローラ144に対応したドライバ116を実装し、ドライバ116に、ドライバ制御情報と被制御装置制御情報（装置対応コマンド）との対応を記述した対応情報リストファイルを実装する。一方、被制御装置140およびその構成機器の設備仕様書、機械設計書、制御S/W設計書、制御盤設計書、コントローラ仕様書などから、XMLを用いて、リソース名変換ファイル126を作成し、ゲートウェイ装置110のデータベース118に実装する。そして、第1の通信部112および第2の通信部120を実装して、上記のようなゲートウェイ装置110を構成することができる。なお、ドライバ116が複数ある場合やデバイスドライバ116aがある場合は、これらをさらに実装する。なお、このゲートウェイ装置110は、ハードにより構成することも可能であり、ソフトウェアにより構成することも可能である。

10

【0058】

以上のような本発明によれば、被制御装置の製造メーカーは、ケーパビリティファイルとドライバを提供するだけで、上位のアプリケーションの仕様を気にすることなく、自社独自の仕様で被制御装置を製造、納入することができる。これにより、被制御装置の仕様の設計自由度が大きくなる。

20

【0059】

ゲートウェイ装置の製造メーカー、または製造システムの使用メーカーは、被制御装置の製造メーカーからケーパビリティファイルとドライバを提供されるため、複雑な開発作業を行うことなく容易にゲートウェイ装置を作製することができ、短い開発・製造納期および安価な開発・製造コストでゲートウェイ装置を作製することができる。

【0060】

また、アプリケーションの製造メーカー、または製造システムの使用メーカーは、被制御装置の製造メーカーからケーパビリティファイルが提供されるため、下位の被制御装置の仕様を気にすることなく、また、複雑な開発作業を行うことなく容易にアプリケーションを製造、納入することができる。これにより、短い開発・製造納期および安価な開発・製造コストでアプリケーションを作製することができる。

30

【0061】

そして、製造システムの使用メーカーでは、特定ドメイン情報（制御要求情報、被制御装置のケーパビリティ、被制御装置の構成機器のケーパビリティ、リソース名変換ファイルなどの被制御装置固有の情報）を交換するだけで、アプリケーションを交換することなく種々の分野の異なる仕様の被制御装置を制御可能である。また、たとえば被制御装置が故障による交換や被制御装置の更新を行った場合においても、特定ドメイン情報ファイルを交換するだけでアプリケーションを交換することなく新たな被制御装置を制御することができる。すなわち、このゲートウェイ装置は、特定ドメイン情報を交換するだけで、種々の装置に対応可能なゲートウェイ装置として機能する。

40

【0062】

したがって、本発明によれば、アプリケーションおよび被制御装置の組み合わせに自由度を持たせることができ、アプリケーションおよび被制御装置の種々の組み合わせに柔軟に対応可能な製造システムおよびこれを実現するゲートウェイ装置が実現されている。

【0063】

実施例

本実施例では、以上のように構成された製造システムにおける被制御装置の制御についての具体的な実施例について図面を参照しながら説明する。なお、本実施例では、上述した図3の構成を用いて説明する。

【0064】

50

まず、制御装置 110 における動作について図 13 を用いて説明する。最初にユーザから制御装置 100 のアプリケーション 102 に、被制御装置 140 に対する制御要求事項である製造要求情報（製造指示）が入力される（ステップ S101）。この製造要求情報としては、たとえば「X」という製品を「タイプ A」という装置を用いて「Y 個」製造する、というような製造指示およびこれに関する属性値が入力される。ここで属性値としては、たとえば加工ラインタイプ、加工装置タイプ、作業名、数量、加工プログラム、使用メモリ容量、などが装置対応コマンドとしてではなく、抽象仕様で入力される。

【0065】

この製造要求情報（製造指示）は、たとえばユーザがスケジューラ等に作業予定を入力すると、該スケジューラ等から自動で作業リストが作成され、該リストが製造要求情報として制御装置 100 にダウンロードされる形態とすることができる。また、たとえばユーザが所定の指示を個別に入力する形態とすることも可能である。これにより、制御装置 100 のアプリケーション 102 のクライアントエンティティ 106 では、所定の制御要求情報が選択、特定される。

10

【0066】

また、ゲートウェイ装置 110 においてドライバが複数存在する場合には、制御要求情報にはゲートウェイ装置 110 においてドライバを選択するためのドライバ選択情報も付随情報として含まれる。したがって、アプリケーション 102 のクライアントエンティティ 106 では、ドライバ選択情報も特定される。すなわち、クライアントエンティティ 106 では、上述した製造要求情報が入力されるとゲートウェイ装置 110 においてドライバが複数存在するか否かを判断する（ステップ S102）。

20

【0067】

そして、ゲートウェイ装置 110 においてドライバが複数存在する場合には（ステップ S102 肯定）、制御要求情報の付随情報として所定のドライバ選択情報を選択、特定する（ステップ S103）。なお、ドライバ情報を個別に選択、特定する形態とすることも可能である。また、ゲートウェイ装置 110 においてドライバが 1 つのみ存在する場合（ドライバが複数存在しない場合）には（ステップ S102 否定）、付随情報としてそのドライバを指定する。

【0068】

つぎに、アプリケーション 102 のクライアントエンティティ 106 では、通信部 108 に対して制御要求情報を送信し、さらに該制御要求情報のゲートウェイ装置 110 への送信を指示する。通信部 108 ではクライアントエンティティ 106 から制御要求情報を受け取り、該制御要求情報をゲートウェイ装置 110 へ送信する（ステップ S104）。また、通信部 108 では、ゲートウェイ装置 110 から返却値が有る場合には、該返却値をクライアントエンティティ 106 に送信する。クライアントエンティティ 106 では、返却値を受け取った場合には、たとえば該返却値に該当するコマンドをディスプレイに表示する等の所定の処理を行う（ステップ S105）。

30

【0069】

そして、クライアントエンティティ 106 では処理が終了したか否かを判断し（ステップ S106）、処理が終了している場合には一連の動作が終了となる。また、処理が終了していない場合には、ステップ S101 に戻り、一連の処理が終了するまで繰り返す。

40

【0070】

つぎに、ゲートウェイ装置 110 における動作を図 14～図 16 を用いて説明する。ゲートウェイ装置 110 では、第 2 の通信部 120 において被制御装置 140 から返却値の入力を確認し（ステップ S111）、被制御装置 140 からの返却値があるか否かを判断する（ステップ S112）。ここで被制御装置 140 からの返却値がある場合には（ステップ S112 肯定）、第 2 の通信部 120 はその返却値をドライバ 116 に送信する。（ステップ S116）。ドライバ 116 では記憶部 124 に記憶された対応情報リストに含まれる逆変換対応情報または逆変換対応情報リストを確認し、該当する返却値の情報が存在するか否かを判断する（ステップ S117）。ここで該当する返却値の情報が存在する

50

場合には（ステップS 1 1 7 肯定）、対応情報リストに含まれる逆変換対応情報または逆変換対応情報リストを用いて返却値をサーバエンティティ 1 1 4 への送信用の返却値情報に逆変換し（ステップS 1 1 8）、変換した返却値情報をサーバエンティティ 1 1 4 に送信する（ステップS 1 1 9）。

【0071】

一方、該当する返却値の情報が、対応情報リストに含まれる逆変換対応情報または逆変換対応情報リストに存在しない場合には（ステップS 1 1 7 否定）、ステップS 1 1 1 に戻る。

【0072】

サーバエンティティ 1 1 4 では、変換した返却値情報を受け取り、記憶部 1 2 2 に記憶された対応情報リストに含まれる逆変換対応情報または逆変換対応情報リストを確認し、該当する返却値情報が存在するか否かを判断する（ステップS 1 2 0）。ここで該当する返却値情報が存在する場合には（ステップS 1 2 0 肯定）、サーバエンティティ 1 1 4 は対応情報リストに含まれる逆変換対応情報または逆変換対応情報リストを用いて、ドライバ 1 1 6 で逆変換された返却値情報を制御装置 1 0 0 への送信用の返却値情報に逆変換し（ステップS 1 2 1）、変換した返却値情報を第 1 の通信部 1 1 2 に送信する（ステップS 1 2 2）。第 1 の通信部 1 1 2 では、サーバエンティティ 1 1 4 から受け取った逆変換した返却値情報を制御装置 1 0 0 に送信する（ステップS 1 2 3）。

【0073】

一方、該当する返却値の情報が、対応情報リストに含まれる逆変換対応情報または逆変換対応情報リストに存在しない場合には（ステップS 1 2 0 否定）、ステップS 1 1 1 に戻る。

【0074】

また、第 2 の通信部 1 2 0 において被制御装置 1 4 0 からの返却値があるか否かを判断した結果、被制御装置 1 4 0 からの返却値がない場合には（ステップS 1 1 2 否定）、つぎに第 1 の通信部 1 1 2 において制御要求情報の入力を確認し（ステップS 1 1 3）、制御要求情報の入力があるか否かを判断する（ステップS 1 1 4）。ここで、制御要求情報の入力がない場合には（ステップS 1 1 4 否定）ステップS 1 1 1 に戻る。一方、制御要求情報の入力がある場合には（ステップS 1 1 4 肯定）、第 1 の通信部 1 1 2 はサーバエンティティ 1 1 4 に制御要求情報を送信する（ステップS 1 1 5）。

【0075】

サーバエンティティ 1 1 4 では、第 1 の通信部 1 1 2 から制御要求情報を受け取ると、制御要求情報とドライバ制御情報との対応情報リストがサーバエンティティ 1 1 4 内の記憶部 1 2 2（メモリ）に存在するか否かを確認する（ステップS 1 2 4）。ここで、対応情報リストがサーバエンティティ 1 1 4 内の記憶部 1 2 2（メモリ）に存在しない場合は（ステップS 1 2 4 否定）、サーバエンティティ 1 1 4 は対応情報リストがデータベース 1 1 8 上に存在するか否かを確認する（ステップS 1 3 4）。その結果、対応情報リストがデータベース 1 1 8 上に存在しない場合には（ステップS 1 3 4 否定）、たとえば、「制御要求情報に対応するドライバ制御情報が無い」とのエラー表示を行う（ステップS 1 3 5）。

【0076】

一方、対応情報リストがデータベース 1 1 8 上に存在するか否かを確認した結果、対応情報リストがデータベース 1 1 8 上に存在する場合には（ステップS 1 3 4 肯定）、サーバエンティティ 1 1 4 は該対応情報リストをデータベース 1 1 8 からサーバエンティティ 1 1 4 の記憶部 1 2 2（メモリ）に読み込む（ステップS 1 3 6）。これにより、サーバエンティティ 1 1 4 における制御要求情報の変換作業が可能となる。

【0077】

ここで、制御要求情報とドライバ制御情報との対応情報リストとしては、たとえば図 1 8 に示すような対応情報リストを用いることができる。図 1 8 に示した制御要求情報とドライバ制御情報との対応情報リストは、制御要求情報リスト ID と、これに対応するドラ

10

20

30

40

50

イバ制御情報（プロセス情報）リスト識別子と、これらの組み合わせをセットで管理する制御要求情報・ドライバ制御情報対応リストIDと、を有している。また、この対応情報とは別に、たとえば図19に示すような制御要求情報リスト、およびたとえば図20に示すようなドライバ制御情報（プロセス情報）リスト一覧を記憶部122に有する。

【0078】

図19に示す制御要求情報リストは、抽象仕様で記述された制御要求情報の詳細と、これに対応する制御要求情報リストIDを含んでいる。制御要求情報としては図19に示すようにたとえば作業ラインや作業装置のタイプである「resource name」、作業内容である「operation name」、製品名である「prduce item」など、装置対応の個別仕様コマンドではなく抽象仕様で記述された製造指示（製造コマンド）が記述される。なお、制御要求情報に基づいて被制御装置140の制御が行われた場合には、これらのそれぞれの項目に対して被制御装置140から動作確認の返却値が戻され、制御確認がなされる。

10

【0079】

また、図20に示すドライバ制御情報（プロセス情報）リスト一覧は、被制御装置140およびその構成機器のケーパビリティに基づいて作成されたドライバ制御情報（プロセス情報）リストの集合である。個々のドライバ制御情報（プロセス情報）リストは、ドライバ制御情報（プロセス情報）リスト識別子ごとに作業手順（シーケンス）である制御コマンドが被制御装置140およびその構成機器のケーパビリティに基づいて作成されたリストであり、ドライバ制御情報（プロセス情報）リストIDと、これに対応するコマンド内容が抽象仕様で記述されている。

20

【0080】

これらのリストを用いることにより、制御要求情報に対応する被制御装置140およびその構成機器のケーパビリティが存在すること、すなわち被制御装置140およびその構成機器において制御要求情報に対応する作業の可否を確実に確認することができる。そして、制御要求情報リストID（たとえばLA001）に対応するドライバ制御情報（プロセス情報）リスト識別子（たとえばLBX001）が存在する場合に、制御要求情報をドライバ制御情報（プロセス情報）に変換することができる。

【0081】

そこで、対応情報リストがサーバエンティティ114内の記憶部122（メモリ）に存在する場合（ステップS124肯定）または上記のステップS136が終了した場合には、サーバエンティティ114では、まず図18に示した制御要求情報とドライバ制御情報との対応情報リストからドライバ制御情報（プロセス情報）リスト識別子を得る（ステップS125）。

30

【0082】

つぎに、サーバエンティティ114は、サーバエンティティ114内の記憶部122（メモリ）に図20に示すドライバ制御情報（プロセス情報）リスト一覧が存在するか否かを確認する（ステップS126）。ここで、ドライバ制御情報（プロセス情報）リスト一覧がサーバエンティティ114内の記憶部122（メモリ）に存在しない場合は（ステップS126否定）、サーバエンティティ114はドライバ制御情報（プロセス情報）リスト一覧がデータベース118上に存在するか否かを確認する（ステップS137）。その結果、ドライバ制御情報（プロセス情報）リスト一覧がデータベース118上に存在しない場合には（ステップS137否定）、たとえば、「識別子に対応するプロセスリストが無い」とのエラー表示を行う（ステップS138）。

40

【0083】

一方、ドライバ制御情報（プロセス情報）リスト一覧がデータベース118上に存在するか否かを確認した結果、ドライバ制御情報（プロセス情報）リスト一覧がデータベース118上に存在する場合には（ステップS137肯定）、サーバエンティティ114は該リスト一覧をデータベース118からサーバエンティティ114の記憶部122（メモリ）に読み込む（ステップS139）。これにより、サーバエンティティ114におけるドライバ制御情報（プロセス情報）リスト識別子からドライバ制御情報（プロセス情報）リ

50

ストIDへの変換作業が可能となる。

【0084】

そこで、ドライバ制御情報（プロセス情報）リスト一覧がサーバエンティティ114内の記憶部122（メモリ）に存在する場合（ステップS126肯定）または上記のステップ139が終了した場合には、サーバエンティティ114では、図20に示したドライバ制御情報（プロセス情報）リスト一覧からドライバ制御情報（プロセス情報）リストIDを得る（ステップS127）。なお、上記のような対応情報リストを用いる代わりに制御要求情報が入力される毎にコンパイラで上記の対応情報リストに相当する結果を得ることも可能である。

【0085】

つぎに、サーバエンティティ114では、ドライバ制御情報（プロセス情報）リストのうち、未だ被制御装置制御情報に変換していないドライバ制御情報（プロセス情報）、すなわちドライバ116に送信していないドライバ制御情報（プロセス情報）を選択し（ステップS128）、該ドライバ制御情報（プロセス情報）が指定するドライバ116を選択する（ステップS129）。そして、選択したドライバ116にドライバ制御情報（プロセス情報）を送信する（ステップS130）。

【0086】

なお、サーバエンティティ114では、このような一連の動作を行うとともに、ドライバ116から逆変換した返却値情報を受け取った場合には、記憶部122に記憶された対応情報リストに含まれる逆変換対応情報または逆変換対応情報リストを確認し、該当する返却値情報が存在するか否かの判断を行う。そして該当する返却値情報が存在する場合には、サーバエンティティ114は対応情報リストに含まれる逆変換対応情報または逆変換対応情報リストを用いて、ドライバ116で逆変換された返却値情報を制御装置100への送信用の返却値情報に逆変換し、第1の通信部112に送信する。

【0087】

ドライバ制御情報（プロセス情報）を受け取ったドライバ116では、該ドライバ116内の記憶部124（メモリ）にドライバ制御情報（プロセス情報）と被制御装置制御情報との対応情報リストが存在するか否かを確認する（ステップS131）。ここで、対応情報リストがドライバ116内の記憶部124（メモリ）に存在しない場合は（ステップS131否定）、ドライバ116は対応情報リストがデータベース118上に存在するか否かを確認する（ステップS140）。その結果、対応情報リストがデータベース118上に存在しない場合には（ステップS140否定）、たとえば、「ドライバ制御情報（プロセス情報）に対応する被制御装置制御情報が無い」とのエラー表示を行う（ステップS141）。

【0088】

一方、対応情報リストがデータベース118上に存在するか否かを確認した結果、対応情報リストがデータベース118上に存在する場合には（ステップS140肯定）、ドライバ116は該対応情報リストをデータベース118からドライバ116の記憶部124（メモリ）に読み込む（ステップS142）。これにより、ドライバ116におけるドライバ制御情報（プロセス情報）の変換作業が可能となる。

【0089】

ここで、ドライバ制御情報（プロセス情報）と被制御装置制御情報との対応情報リストとしては、図21に示すような対応情報リストを用いることができる。図21に示したドライバ制御情報（プロセス情報）と被制御装置制御情報との対応情報リストは、ドライバ制御情報（プロセス情報）リストIDと、これに対応する被制御装置制御情報リストIDと、これらの組み合わせをセットで管理するドライバ制御情報・被制御装置制御情報対応リストIDと、を有している。また、この対応情報とは別に、たとえば図22に示すようなドライバ制御情報（プロセス情報）リスト、およびたとえば図23に示すような被制御装置制御情報リストを記憶部124に有する。

【0090】

図 2 2 に示すドライバ制御情報（プロセス情報）リストは、図 2 0 に示した個々のドライバ制御情報（プロセス情報）リストからドライバ制御情報（プロセス情報）リスト識別子を除いたものである。すなわち、リスト識別子ごとに作業手順（シーケンス）である制御コマンドが被制御装置 1 4 0 およびその構成機器のケーブルリティに基づいて作成されたリストであり、ドライバ制御情報（プロセス情報）リスト ID（たとえば P 1、P 2 など）と、これに対応するコマンド内容が抽象仕様で記述されている。

【0091】

また、図 2 3 に示す被制御装置制御情報リストは、被制御装置制御情報リスト ID とこれに対応する制御コマンドが装置対応コマンドである個別仕様の制御コマンドとして記述されている。

10

【0092】

これらのリストを用いることにより、ドライバ制御情報（プロセス情報）に対応する被制御装置 1 4 0 およびその構成機器のケーブルリティが存在すること、すなわち被制御装置 1 4 0 およびその構成機器においてドライバ制御情報（プロセス情報）に対応する作業の可否を確実に確認することができる。そして、ドライバ制御情報（プロセス情報）リスト ID（たとえば P 1）に対応する被制御装置制御情報リスト ID（たとえば L C 0 0 1）が存在する場合に、ドライバ制御情報（プロセス情報）を装置対応の個別仕様の被制御装置制御情報に変換することができる。すなわち、制御要求情報はドライバ 1 1 6 において初めて装置対応の個別仕様の情報に変換される。

【0093】

20

そこで、対応情報リストがドライバ 1 1 6 内の記憶部 1 2 4（メモリ）に存在する場合（ステップ S 1 3 1 肯定）または上記のステップ 1 4 2 が終了した場合には、ドライバ 1 1 6 では、図 2 1 に示したドライバ制御情報（プロセス情報）と被制御装置制御情報との対応情報リストから、装置対応の制御コマンドである被制御装置制御情報を得る（ステップ S 1 3 2）。なお、上記のような対応情報リストを用いる代わりにドライバ制御情報（プロセス情報）が入力される毎にコンパイラで上記の対応情報リストに相当する結果を得ることも可能である。

【0094】

つぎに、ドライバ 1 1 6 ではドライバ制御情報（プロセス情報）が全て被制御装置制御情報に変換されたか否かを判断し（ステップ S 1 3 3）、未変換のドライバ制御情報（プロセス情報）が有る場合には（ステップ S 1 3 3 否定）、ステップ S 1 2 8 に戻る。また、ドライバ制御情報（プロセス情報）が全て被制御装置制御情報に変換された場合には（ステップ S 1 3 3 肯定）、ドライバ 1 1 6 は第 2 の通信部 1 2 0 に被制御装置制御情報を送信する（ステップ S 1 4 3）。そして第 2 の通信部 1 2 0 は、被制御装置 1 4 0 に被制御装置制御情報を送信する（ステップ S 1 4 4）。

30

【0095】

なお、ドライバ 1 1 6 では、このような一連の動作を行うとともに、被制御装置 1 4 0 返却値情報を受け取った場合には、記憶部 1 2 4 に記憶された対応情報リストに含まれる逆変換対応情報または逆変換対応情報リストを確認し、該当する返却値情報が存在するかどうかの判断を行う。そして該当する返却値情報が存在する場合には、ドライバ 1 1 6 は対応情報リストに含まれる逆変換対応情報または逆変換対応情報リストを用いて、返却値をサーバエンティティ 1 1 4 への送信用の返却値情報に逆変換し、サーバエンティティ 1 1 4 に送信する。

40

【0096】

つぎに、被制御装置 1 4 0 における動作を図 1 7 を用いて説明する。被制御装置 1 4 0 では、通信回線 1 5 6 を介して第 2 の通信部 1 2 0 から装置対応の制御コマンドに変換された被制御装置制御情報を通信部 1 4 2 が受け取る（ステップ S 1 5 0）。通信部 1 4 2 は被制御装置制御情報を受け取ると、該被制御装置制御情報が自被制御装置 1 4 0 に対応するドライバから送信された情報であるか否かを確認する（ステップ S 1 5 1）。ここで、受け取った被制御装置制御情報が、自被制御装置 1 4 0 に対応するドライバから送信さ

50

れた情報ではない場合は（ステップS 1 5 1 否定）、ステップS 1 5 0に戻る。

【0097】

また、受け取った被制御装置制御情報が、自被制御装置140に対応するドライバから送信された情報である場合は（ステップS 1 5 1 肯定）、通信部142はつぎに該被制御装置制御情報がコントローラ144を介して制御する制御コマンドであるか否かを確認する（ステップS 1 5 2）。ここで、受け取った被制御装置制御情報がコントローラ144を介して制御する制御コマンドでない場合は（ステップS 1 5 2 否定）、通信部142はさらに該被制御装置制御情報が通信回線152に載せることができる制御コマンドであるか否かを確認する（ステップS 1 5 7）。

【0098】

ここで、受け取った被制御装置制御情報が通信回線152に載せることができる制御コマンドでない場合には（ステップS 1 5 7 否定）、通信部142は制御コマンドエラーと判断し（ステップS 1 5 9）、ステップS 1 5 0に戻る。一方、受け取った被制御装置制御情報が通信回線152に載せることができる制御コマンドである場合には（ステップS 1 5 7 肯定）、通信部142は被制御装置制御情報を通信回線152に載せ（ステップS 1 5 8）、対応する被制御装置140の構成機器（146等）の制御が行われる。

【0099】

また、受け取った被制御装置制御情報がコントローラ144を介して制御する制御コマンドである場合は（ステップS 1 5 2 肯定）、通信部142は被制御装置制御情報をコントローラ144に送信する（ステップS 1 5 3）。コントローラ144においては、被制御装置制御情報を通信回線152に合致するプロトコルに変換し、対応する被制御装置140の構成機器（146等）、の制御を行う（ステップS 1 5 4）。

【0100】

そして、コントローラ144では、被制御装置140の構成機器からの返却値があるか否かを確認する（ステップS 1 5 5）。ここで、返却値がある場合には（ステップS 1 5 5 肯定）、通信部142を介してゲートウェイ装置110の第2の通信部120に返却値を送信する（ステップS 1 6 0）。一方、返却値がない場合には（ステップS 1 5 5 否定）、コントローラ144は被制御装置140の構成機器の制御が終了したか否かを確認する（ステップS 1 5 6）。ここで、被制御装置140の構成機器（146等）の制御が終了していない場合には（ステップS 1 5 6 否定）、ステップS 1 5 4に戻る。また、被制御装置140の構成機器（146等）の制御が終了した場合には（ステップS 1 5 6 肯定）、一連の被制御装置140の制御が終了する。

【0101】

また、被制御装置の構成の一例としては、たとえば素材加工ラインをイメージした構成であり、図24に示すように通信部142と、コントローラ144と、被制御装置250を構成する加工素材を製造ラインに搬入するライン搬入用ロボット252と、製造ライン上において加工素材を搬送する搬送装置254と、加工素材の荒加工を行う加工機A256と、加工素材の仕上げ加工を行う加工機B258と、を備えた構成も挙げられる。なお、コントローラ144と、ライン搬入用ロボット252と、搬送装置254と、加工機A256と、加工機B258とは、被制御装置250内のネットワーク260により接続されている。

【産業上の利用可能性】

【0102】

以上のように、本発明にかかる製造システムは、制御対象設備と、これらの制御対象設備の運用のための基本情報を処理して運転スケジュールを作成しながらこれらの制御対象設備を統括的に運転制御する制御装置とが、ネットワークで接続された構成を有するフレキシブル生産システムに適している。

【図面の簡単な説明】

【0103】

【図1】従来の製造システムの概要を説明する図である。

10

20

30

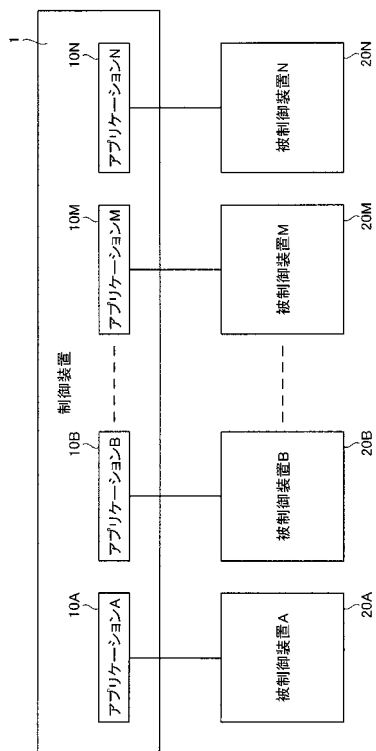
40

50

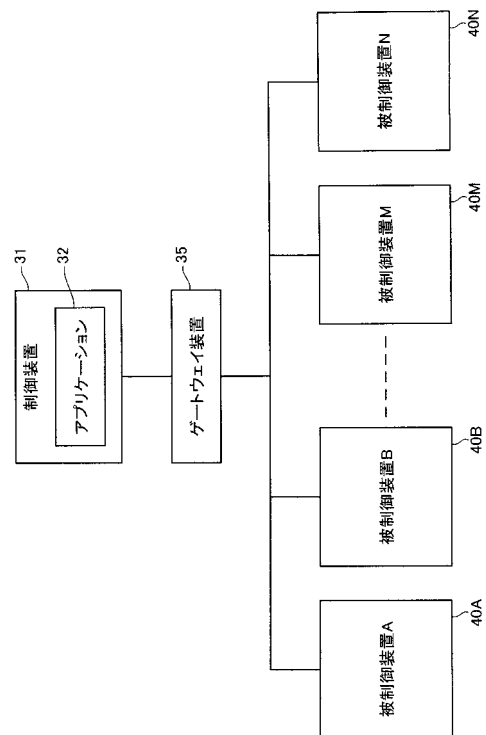
- 【図 2】本発明の製造システムの概要を説明する図である。
- 【図 3】本発明の製造システムの概要を説明する図である。
- 【図 4】ドライバとコントローラとの関係を説明する図である。
- 【図 5】ドライバとコントローラとの関係を説明する図である。
- 【図 6】ドライバとコントローラとの関係を説明する図である。
- 【図 7】本発明の製造システムの全体的な動作手順を示すフローチャートである。
- 【図 8】制御経路を説明する図である。
- 【図 9】制御経路を説明する図である。
- 【図 10】制御経路を説明する図である。
- 【図 11】抽象インターフェイス記述言語による記述例を示す図である。 10
- 【図 12】IDLによる記述例を示す図である。
- 【図 13】本発明の製造システムにおける制御装置の動作手順を示すフローチャートである。
- 【図 14】本発明の製造システムにおけるゲートウェイ装置の動作手順を示すフローチャートである。
- 【図 15】本発明の製造システムにおけるゲートウェイ装置の動作手順を示すフローチャートである。
- 【図 16】本発明の製造システムにおけるゲートウェイ装置の動作手順を示すフローチャートである。
- 【図 17】本発明の製造システムにおける被制御装置の動作手順を示すフローチャートである。 20
- 【図 18】制御要求情報とドライバ制御情報との対応情報リストを示す図である。
- 【図 19】制御要求情報リストを示す図である。
- 【図 20】ドライバ制御情報（プロセス情報）リスト一覧を示す図である。
- 【図 21】ドライバ制御情報（プロセス情報）と被制御装置制御情報との対応情報リストを示す図である。
- 【図 22】ドライバ制御情報（プロセス情報）リストを示す図である。
- 【図 23】被制御装置制御情報リストを示す図である。
- 【図 24】被制御装置の他の構成例を示す図である。
- 【符号の説明】 30
- 【0104】
- 100 制御装置
 - 102 アプリケーション
 - 104 記憶部
 - 106 クライアント—エンティティ
 - 108 通信部
 - 110 ゲートウェイ装置
 - 112 第1の通信部
 - 114 サーバーエンティティ
 - 116 ドライバ 40
 - 118 データベース
 - 120 第2の通信部
 - 122 記憶部
 - 124 記憶部
 - 126 リソース名変換ファイル
 - 140 被制御装置
 - 142 通信部
 - 144 コントローラ
 - 146 ロボット
 - 148 CNC 50

- 150 サーボ
- 152 ネットワーク
- 154 通信回線
- 156 通信回線

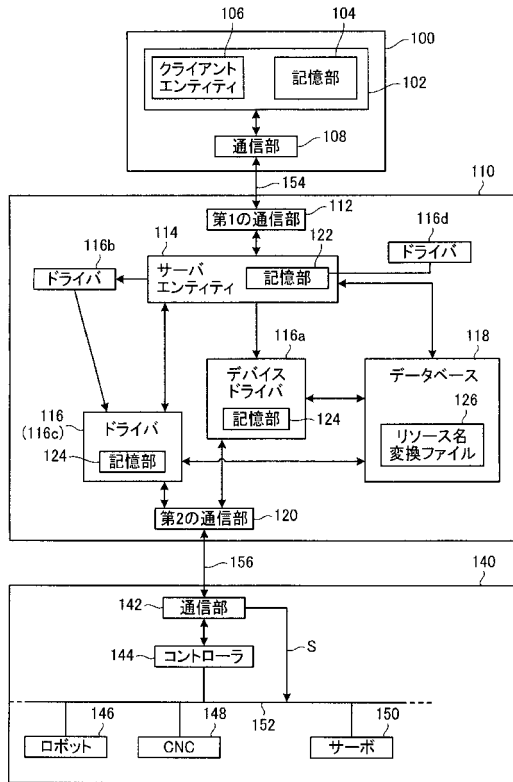
【図1】



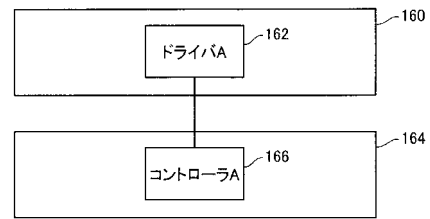
【図2】



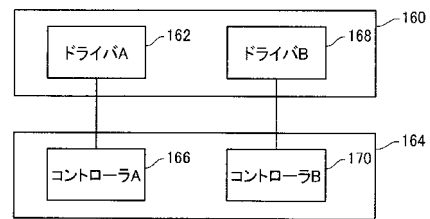
【図 3】



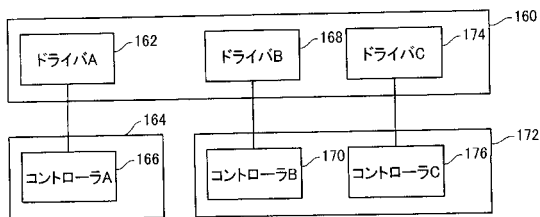
【図 4】



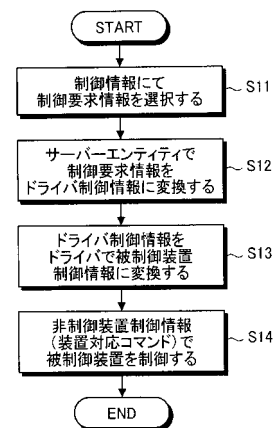
【図 5】



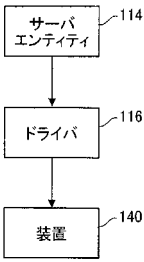
【図 6】



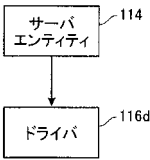
【図 7】



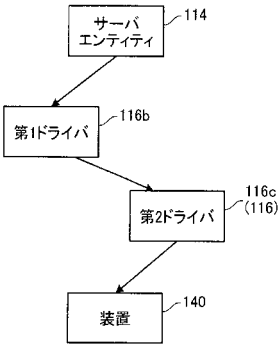
【図 8】



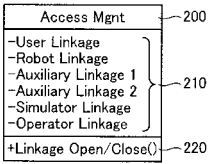
【図 10】



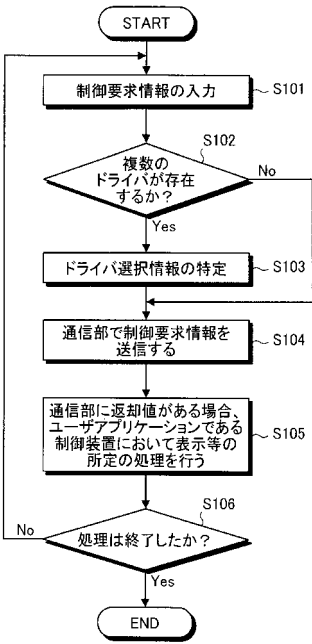
【図 9】



【図 11】



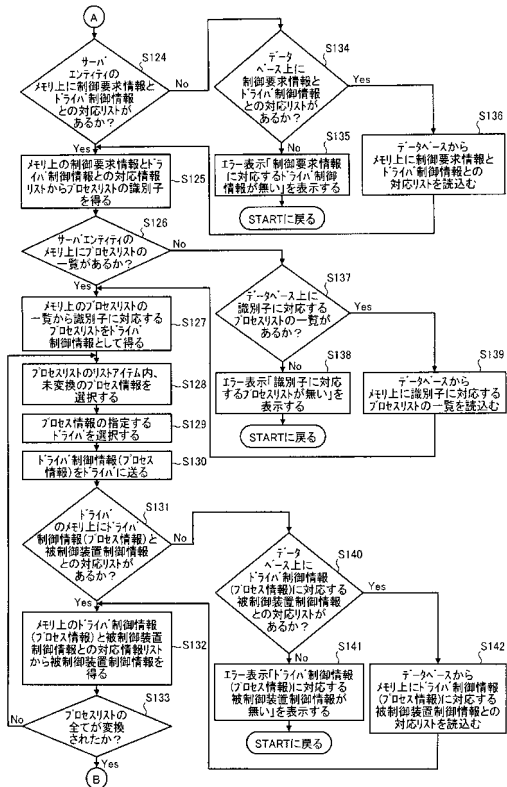
【図 13】



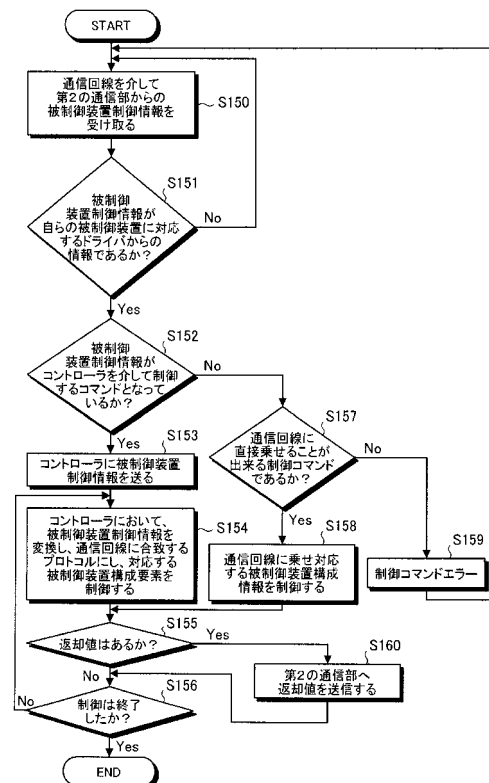
【図 12】

```
Interface Gateway_x {
    attribute string vender_name;
    attribute string system_name;
    attribute long xxx;
    ...
    void func1(in long parameter11,character parameter12);
    void func2(in boolean parameter21);
    long func3(out long parameter31,...);
    void set_attribute_xxx(in...);
    long get_attribute_xxx(out...);
}
```

【図 15】



【図 17】



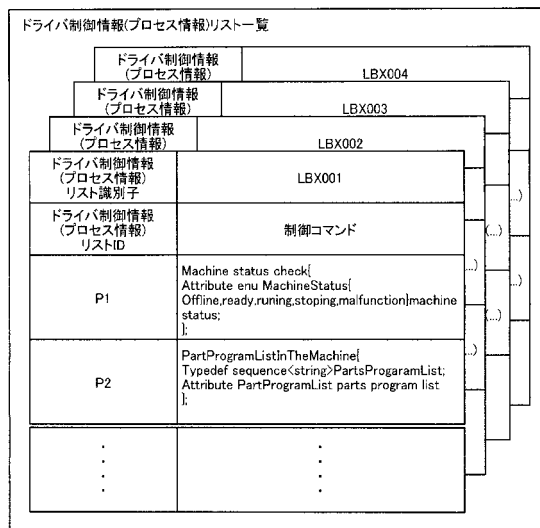
【図 18】

制御要求情報とドライバ制御情報との対応情報リスト		
制御要求情報・ドライバ制御情報対応リストID	制御要求情報リストID	ドライバ制御情報(プロセス情報)リスト識別子
LX001	LA001	LBX001
LX002	LA002	LBX002
⋮	⋮	⋮

【図 19】

制御要求情報リスト	
制御情報対応リストID	制御要求情報
LA001	<pre> <resource name="シャフトライン"/> <resource name="マシセン1"/><part of resource="シャフトライン"/> </resource> <operation name="作業A-012345-1"> <produce item="製品 P012345"> <quantity value="12"/> <duration value="180"/> <start><time value="2004-10T09:00:00"/></start> <end><time value="2004-10T09:36:00"/></end> </operation> </pre>
LA002	<pre> <resource name="ブラケットライン"/> <resource name="ブラケットA1"> <part of resource="ブラケットライン"/> </resource> <operation name="作業A-012345-2"> <produce item="製品 P010045"> <quantity value="12"/> </pre>
⋮	⋮

【図 20】



【図 21】

ドライバ制御情報と被制御装置制御情報との対応情報リスト		
ドライバ制御情報・被制御装置制御情報対応リストID	ドライバ制御情報(プロセス情報)リストID	被制御装置制御情報リストID
LY001	P1	LC001
LY002	P2	LC002
⋮	⋮	⋮

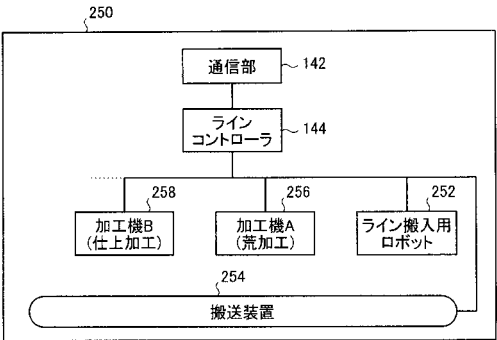
【図 22】

ドライバ制御情報(プロセス情報)リスト	
ドライバ制御情報(プロセス情報)リストID	制御コマンド
P1	<pre> Machine status check{ Attribute enu MachineStatus{ Offline.ready.runing.stopping.malfunction}machine status; }; </pre>
P2	<pre> PartProgramListInTheMachine{ Typedef sequence<string>PartsProgramList; Attribute PartProgramList parts program list }; </pre>
⋮	⋮

【図 2 3】

被制御装置制御情報リスト	
被制御装置制御情報ID	制御コマンド
LC001	MCheck()
LC002	PPListIn(...)
⋮	⋮

【図 2 4】



フロントページの続き

(58)調査した分野(Int.Cl., DB名)

H04L 12/28-46

H04Q 9/00