

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号
特許第5133473号
(P5133473)

(45) 発行日 平成25年1月30日 (2013. 1. 30)

(24) 登録日 平成24年11月16日 (2012. 11. 16)

(51) Int. Cl.

F I

G O 5 B 13/02 (2006. 01)

G O 5 B 15/02 (2006. 01)

G O 5 B 13/02 D

G O 5 B 15/02 M

請求項の数 24 (全 23 頁)

(21) 出願番号	特願2000-72787 (P2000-72787)	(73) 特許権者	512132022
(22) 出願日	平成12年3月15日 (2000. 3. 15)		フィッシャー・ローズマウント システムズ、インコーポレイテッド
(65) 公開番号	特開2000-293202 (P2000-293202A)		アメリカ合衆国 テキサス州 7 8 6 8 1
(43) 公開日	平成12年10月20日 (2000. 10. 20)		ラウンド ロック ウェスト ルイス
審査請求日	平成19年2月6日 (2007. 2. 6)		ヘナ ブルバード 1 1 0 0 ビルディング 1
審判番号	不服2011-2112 (P2011-2112/J1)	(74) 代理人	110000556
審判請求日	平成23年1月28日 (2011. 1. 28)		特許業務法人 有古特許事務所
(31) 優先権主張番号	09/268219	(74) 復代理人	100145621
(32) 優先日	平成11年3月15日 (1999. 3. 15)		弁理士 高田 聡
(33) 優先権主張国	米国 (US)	(72) 発明者	テレンス エル. プレビンス
			アメリカ合衆国 7 8 6 8 1 テキサス
			ラウンド ロック カーメル ドライブ
			3 8 0 1
			最終頁に続く

(54) 【発明の名称】 自動チューナおよび制御要素をチューニングする方法

(57) 【特許請求の範囲】

【請求項 1】

通信ネットワークによって通信結合された第 1 および第 2 の装置と、第 1 の装置内に位置する制御要素とを有するプロセス制御システムにおいて用いられる自動チューナであって、

第 1 の装置内に位置した第 1 のチューニング要素を含み、前記第 1 のチューニング要素は、

制御要素に通信結合され、チューニング手続きの間に制御要素の動作を制御するチューニングパラメータを有する制御要素コントローラと、

チューニング手続きの間の前記プロセス制御システムに関連するデータを集めるデータ収集ユニットと、

データ収集ユニットで集められたデータからプロセス特性を計算するプロセス特性計算機と、

通信ネットワークによって第 2 の装置にプロセス特性計算機で計算されたプロセス特性を示すプロセス特性情報を伝えるコミュニケータとを含み、

前記自動チューナはさらに、第 2 の装置内に位置し、通信ネットワークを介して第 1 のチューニング要素に通信結合された第 2 のチューニング要素を含み、

前記第 2 のチューニング要素は、第 1 のチューニング要素がデータを集め、当該集めたデータからプロセス特性を示すプロセス特性情報を計算し、前記プロセス特性を前記第 2 のチューニング要素に伝えるチューニングの動的データキャプチャ段階とは時間的に分離

10

20

したチューニングの規則アプリケーション段階の間に、プロセス特性情報から制御要素コントローラに出力されるチューニングパラメータの値を決定し、

制御要素コントローラは、チューニングパラメータの値に基づいて制御要素の動作を制御する自動チューナ。

【請求項 2】

プロセス特性情報は、集められたデータを含む、請求項 1 に記載の自動チューナ。

【請求項 3】

前記第 2 のチューニング要素は、特定の種類の制御要素のためのチューニングパラメータの値を決定するのに用いられる一組の規則を含む、請求項 1 に記載の自動チューナ。

【請求項 4】

一組の規則は、比例・積分・微分制御要素のためのチューニングパラメータの値を決定するように構成される、請求項 3 に記載の自動チューナ。

【請求項 5】

一組の規則は、比例・積分制御要素のためのチューニングパラメータの値を決定するように構成される、請求項 3 に記載の自動チューナ。

【請求項 6】

一組の規則は、ファジィ論理制御要素のためのチューニングパラメータの値を決定するように構成される、請求項 3 に記載の自動チューナ。

【請求項 7】

一組の規則は、ニューラルネットワークを用いてチューニングパラメータの値を決定するように構成される、請求項 3 に記載の自動チューナ。

【請求項 8】

第 2 のチューニング要素は、複数組の規則を含み、該複数組の規則の各々は、異なる種類の制御要素のためのチューニングパラメータの値を決定するように構成される、請求項 1 に記載の自動チューナ。

【請求項 9】

第 1 の装置はフィールドバス装置であり、前記第 1 のチューニング要素は、フィールドバスプロトコルに従う自動チューナ機能ブロックである、請求項 1 に記載の自動チューナ。

【請求項 10】

前記第 2 のチューニング要素は、決定されたチューニングパラメータの値を制御要素コントローラに送る、請求項 1 に記載の自動チューナ。

【請求項 11】

前記第 2 のチューニング要素は、プロセス制御システムと関連する集中化コントローラ内に位置している、請求項 1 に記載の自動チューナ。

【請求項 12】

前記第 2 のチューニング要素は、プロセス制御システムと関連するユーザワークステーション内に位置している、請求項 1 に記載の自動チューナ。

【請求項 13】

第 3 の装置内に位置した第 3 のチューニング要素をさらに含み、第 3 の装置はさらなる制御要素を含み、前記第 3 のチューニング要素は、

第 3 の装置内でさらなる制御要素に通信結合され、さらなるチューニング手続きの間にさらなる制御要素の動作を制御する第 2 のチューニングパラメータを有する第 2 の制御要素コントローラと、

さらなるチューニング手続きの間に生成されたデータを集める第 2 のデータ収集ユニットと、

前記制御要素コントローラのための前記第 2 のチューニングパラメータの値を決定するのに用いるために、通信ネットワークによって第 2 の装置にさらなる制御要素と関連するプロセス特性情報を伝える第 2 のコミュニケータとを含む、請求項 1 に記載の自動チューナ。

10

20

30

40

50

【請求項 1 4】

通信ネットワークによって通信結合された第 1 および第 2 の装置と、第 1 および第 2 の装置の各々内でソフトウェアを実行することが出来るプロセッサと、第 1 および第 2 の装置内の各々内のメモリと、第 1 の装置内に位置する制御要素とを有するプロセス制御システムにおいてチューニング手続きを実行するのに用いられる自動チューナであって、

第 1 の装置のメモリ内に記憶可能且つ第 1 の装置のプロセッサ上で実行可能なチューニング機能ブロックソフトウェアを含み、前記チューニング機能ブロックソフトウェアは、
チューニング手続きの間に第 1 の装置内の制御要素の動作をチューニングパラメータを用いて制御する制御ルーチンと、

チューニング手続きの間に得られた前記プロセス制御システムに関連するデータを集め、第 1 の装置のメモリ内に集められたデータを記憶するデータ収集ルーチンと、

前記データ収集ルーチンで集められたデータからプロセス特性を計算するプロセス特性計算ルーチンと、

通信ネットワークによって前記プロセス特性計算ルーチンで計算されたプロセス特性を示すプロセス特性情報を第 2 の装置に伝える通信ルーチンとを含み、

前記自動チューナはさらに、

第 2 の装置のメモリ内に記憶可能且つチューニング手続きの間にデータを集め、当該集めたデータからプロセス特性を示すプロセス特性情報を計算する動的データキャプチャ段階とは時間的に分離したチューニングの規則アプリケーション段階の間に、プロセス特性情報から前記制御ルーチンのためのチューニングパラメータの値を決定し、当該チューニングパラメータの値を前記制御ルーチンに出力する第 2 の装置のプロセッサ上で実行可能なチューニングパラメータ計算ソフトウェアを含み、

チューニング機能ブロックソフトウェアは、チューニングパラメータの値に基づいて制御要素の動作を制御する自動チューナ。

【請求項 1 5】

前記チューニングパラメータ計算ソフトウェアは、計算されたプロセス特性から特定の種類の制御要素のためのチューニングパラメータの値を決定するのに用いられる一組の規則を含む、請求項 1 4 に記載の自動チューナ。

【請求項 1 6】

一組の規則は、比例・積分・微分制御要素のためのチューニングパラメータの値を決定するように構成される、請求項 1 5 に記載の自動チューナ。

【請求項 1 7】

一組の規則は、ファジイ論理制御要素のためのチューニングパラメータの値を決定するように構成される、請求項 1 5 に記載の自動チューナ。

【請求項 1 8】

一組の規則は、ニューラルネットワークを用いてチューニングパラメータの値を決定するように構成される、請求項 1 5 に記載の自動チューナ。

【請求項 1 9】

第 1 の装置はフィールドバスフィールド装置であり、前記チューニング機能ブロックソフトウェアは、フィールドバスプロトコルに従う自動チューナ機能ブロックである、請求項 1 4 に記載の自動チューナ。

【請求項 2 0】

通信ネットワークによって通信結合された第 1 の装置および第 2 の装置を有するプロセス制御システムの第 1 の装置内に位置し、チューニングパラメータを有する制御要素をチューニングする方法であって、

第 1 の装置内に記憶された制御手続きを実行して制御要素の動作を制御し、それによってチューニング手続きを実行するステップと、

第 1 の装置においてチューニング手続きの間に得られた、第 1 の装置の制御出力に関連するデータを集めるステップと、

集められたデータからプロセス特性を決定するステップと、

10

20

30

40

50

通信ネットワークによって、集められたデータから展開され、決定されたプロセス特性を示すプロセス特性情報を第2の装置に伝えるステップと、

第2の装置において、チューニング手続きの間にデータを集め、当該集めたデータからプロセス特性を示すプロセス特性情報を計算する動的データキャプチャ段階とは時間的に分離したチューニングの規則アプリケーション段階の間に、プロセス特性情報から制御要素のためのチューニングパラメータの値を決定するステップとを含み、

チューニング手続きを実行するステップは、決定されたチューニングパラメータの値に基づいて制御要素の動作を制御する、制御要素をチューニングする方法。

【請求項21】

プロセス特性情報は集められたデータを含む、請求項20に記載の制御要素をチューニングする方法。

10

【請求項22】

前記チューニングパラメータの値を決定するステップは、第2の装置内で特定の種類の制御要素のためのチューニングパラメータを決定するのに用いられる一組の規則を記憶するステップと、記憶された一組の規則を用いて、プロセス特性情報に基づいてチューニングパラメータの値を決定するステップとを含む、請求項20に記載の制御要素をチューニングする方法。

【請求項23】

前記チューニングパラメータの値を決定するステップは、複数組の規則を記憶するステップを含み、該複数組の規則の各々は異なる種類の制御要素のためのチューニングパラメータの値を決定するように構成され、前記決定するステップはさらに、チューニングされている制御要素の種類に依って複数組の規則のうちの異なるものを用いるステップを含む、請求項20に記載の制御要素をチューニングする方法。

20

【請求項24】

前記チューニングパラメータの値を決定するステップの後に、通信ネットワークによってチューニングパラメータの値を制御要素に送るステップをさらに含む、請求項20に記載の制御要素をチューニングする方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

30

本発明は一般にプロセス制御ネットワークに関し、より特定的にはプロセス制御環境中に分散された制御要素に対し自動チューニングを行なう装置および方法に関する。

【0002】

【従来の技術および発明が解決すべき課題】

化学処理、石油処理、または他の処理で用いられるもののように、プロセス制御ネットワークは一般に、例えば、弁ポジション、スイッチ、センサ（温度、圧力、および流量センサ）などのひとつ以上のフィールド装置に通信結合された集中型プロセスコントローラを含んでいた。これらのフィールド装置はプロセス内の物理的な制御機能（弁の開閉のような）を行なっても、プロセスの動作を制御するのに用いられるプロセス内の測定値の測定を行なっても、プロセス内で他の所望の機能を実行してもよい。プロセスコントローラは、例えば、フィールド装置へそしてそこから4ないし20（以下、〇ー〇のように表す）mA（ミリアンペア）の信号を搬送し得るひとつ以上のアナログ信号線またはバスによってフィールド装置に接続されていた。一般に、プロセスコントローラは、ひとつ以上のフィールド装置により測定された測定値、および／またはフィールド装置に関する他の情報を示す信号を受け取り、この情報を用いて典型的には複雑な制御ルーチンを実行し、ついで制御信号を生成し、この制御信号はアナログ信号バスによってフィールド装置に送られて、それによってプロセスの動作を制御する。

40

【0003】

最近、プロセス制御環境内でフィールドベースのデジタル通信を実現するために、プロセス制御産業内で動きがあった。例えば、プロセス制御産業は、HART（登録商標）、

50

PROFIBUS（登録商標）、WORLDFIP（登録商標）、Device-Net（登録商標）、およびCANプロトコルなど、多くの標準的、開放的、デジタルのまたは組み合わされたデジタルおよびアナログの通信プロトコルを開発してきた。これらのデジタル通信プロトコルによって、一般に、より多くのフィールド装置が特定のバスに接続されるようになり、フィールド装置とコントローラとの間でより多くのかつ高速の通信を支援し、および／または、フィールド装置が、フィールド装置自身の状態および機器構成に関する情報のようなより多くのかつ様々な種類の情報をプロセスコントローラに送ることが可能となる。さらに、これらの標準的なデジタルプロトコルによって、様々な製造業者により製造されたフィールド装置が同じプロセス制御ネットワーク内で一緒に用いられるようになる。

10

【0004】

現在、プロセス制御を非集中化（分散化）し、それによってプロセスコントローラを単純化するために、プロセス制御産業内で動きがある。非集中化された制御は、弁ポジション、送信機などのようなフィールド装着プロセス制御装置に、典型的には機能ブロックまたは制御ブロックと称されるものを用いてひとつ以上のプロセス制御機能を実行させることにより、そして他の制御機能を実行する際に他のプロセス制御装置（または機能ブロック）により用いられるバス構造によってデータを伝えることにより得られる。これらの制御機能を実行するために、各プロセス制御装置は典型的には、標準的なかつ開放的な通信プロトコルを用いて他のプロセス制御装置と通信する能力だけでなくひとつ以上の機能ブロックを実行する性能（capability）を有するマイクロプロセッサを含む。このように、フィールド装置は、集中化されたプロセスコントローラの介在なしに、プロセス制御ネットワーク内で相互接続されて、互いと通信し、制御ループを形成するひとつ以上の制御機能を実行することが可能である。Foundation（登録商標）Fieldbus（以後“フィールドバス”）プロトコルとして知られているフィールドバスファウンデーションにより今現在宣伝されている全デジタル、2線式バスプロトコルは、様々な製造業者により製造された装置が標準的なバスによって互いに相互動作し通信して、プロセス内で非集中化された制御を達成する開放的な通信プロトコルである。

20

【0005】

ひとつ以上の集中化されたコントローラ内に位置する全体のプロセス制御ルーチン（例えば制御ルーチンの機能ブロック全て）またはその部分を有する先行技術のシステム内で制御ブロックまたは制御ループをチューニングすることはかなり単純である。なぜならば全体のチューニングは集中化されたコントローラ内に記憶可能であるからだ。このような集中化された制御ルーチンの制御ループのチューニングが望まれる場合、集中化されたコントローラ内の別個のチューニングルーチンは、誘導された発振手続きのようなチューニング手続きを経て、比例・積分（PI）または比例・積分・微分（PID）制御ブロックのような適切な制御ブロックに、プロセスまたはループの予め定められた特性を強制的に決定させる。チューニング手続きのこの動的データキャプチャ（取りこみ）段階の間、チューニングルーチンはループにより生成されたデータを集め、このデータは、正規動作毎に、集中化されたコントローラに送られているが、チューニングルーチンはこのデータから、プロセスの極限利得、時定数などのようなひとつ以上のプロセス特性を決定する。いったん所望のプロセス特性が計算されると、チューニングルーチンは、制御ブロックまたは制御ループのための新しいチューニングパラメータを決定するために、計算されたプロセス特性を用いて一組の規則または他のアルゴリズムを応用する。このステップは通常、チューニング手続きの規則アプリケーション段階と称される。その後、チューニングルーチンは、新しいチューニングパラメータを制御ブロック（または制御ループ）に送り、チューニング手続きが完了する。なぜなら、集中化されたプロセス制御システムにおいて、制御機能の全てはコントローラ内に位置し、チューニングのために必要なデータ全てはプロセスの正規動作の間にコントローラに与えられるので、チューニングルーチンは、制御ブロックに直接アクセスし、チューニングルーチンを実行するために必要なデータに直接アクセスする。

30

40

50

【0006】

PI、PID、ファジイ論理などの制御ブロックのような制御ブロックまたは制御要素がプロセス制御ネットワーク中に分散されて位置している非集中化された通信プロトコルを用いると、制御ブロック（またはこれらのブロックが動作している制御ループ）をチューニングするのは難しい。なぜなら、制御ブロックは、チューニングルーチンが典型的には記憶されている集中化コントローラ（または他の装置）から離れて位置しているからである。分散化プロセス制御環境においてチューニングを実行するのに用いられる当該技術で公知のシステムにおいては、全体のチューニング手続きは集中化プロセスコントローラ内にあるままである。しかしながら、このシステムは、高速チューニングを行なうことはできない。なぜなら、それは、バスネットワーク（プロセス内で他の通信を与えている）によって通信を行なってチューニングルーチンの間に生じたデータを受信しなければならないからであり、そして、残念なことに、データの量（またはチューニングルーチンがこのデータを受信することが出来る速度）はバススループットの制約により制限されるからである。さらに、バス通信は別個のバスコントローラにより制御されているのであって、チューニングルーチンにより制御されているわけではないので、チューニングルーチンは、チューニング手続きの様々なセグメントを開始、停止、および実行するためにチューニング制御信号が制御ブロックに送られる正確な時間を厳密に制御することはできない。これは、チューニング制御ルーチンは、チューニング手続きのタイミングに対し厳密な制御を有しないということを意味し、このことは不正確な結果につながる。

10

【0007】

分散化プロセス制御環境内でチューニングを与える別の当該技術で公知のシステムにおいては、全体のチューニングルーチンは、チューニングされるべき制御ブロック（PID機能ブロックのような）と同じ装置内に配置され、事実、制御ブロックの機能性内に実際に組み込まれている。このシステムは、正確にチューニング手続きのタイミングを制御し、所望の速度でデータを集めることが出来るが（なぜならチューニングルーチンはバスによって制御ブロックと通信する必要はないので）、チューニングルーチンは制御ブロックと一緒にかつ同時にコンパイルされなければならない、これは、自動チューニングルーチンの機能性が制御ループの正規動作の間に比較的あまり使用されずとも、プロセスの正規動作の間に制御ブロックを用いることに関連したオーバーヘッド（例えばタイミング、処理、メモリなどの要件）を増大させる。さらに、制御ブロックが各制御ブロックの自動チューニングを可能化するために位置している各々異なる装置内に完全な自動チューニングルーチンが配置されなければならない、これはプロセス制御システムに不必要な冗長性を付加し、プロセス制御システムの費用を増大させる。

20

30

【0008】

本発明は、分散化制御機能を有するプロセス制御ネットワーク内で制御要素をチューニングするのに用いられる自動チューナ及びチューニング方法を提供することを目的としている。

【0009】

【課題を解決するための手段】

分散化制御機能を有するプロセス制御ネットワークにおいて制御要素（制御ブロックのような）をチューニングするのに用いられる自動チューナは、制御要素が動作しているコントローラまたはフィールド装置内に位置した第1のチューニング要素と、第1のチューニング要素が位置するコントローラまたはフィールド装置に接続されたオペレータワークステーション、パソコン、または集中化コントローラのような異なる装置に位置する第2の装置とを含み、第2のチューニング要素はバスまたは他の通信ネットワークによって第1のチューニング要素と通信する。第1のチューニング要素は、自動チューニング手続きの動的データキャプチャ段階の間に制御ブロックの動作を制御し、自動チューニング手続きのこの段階の間にデータを集め、好ましくは、集められたデータからひとつ以上のプロセス（例えばループ）特性を計算する。第1のチューニング要素は、自動チューナの第2のチューニング要素に通信ネットワークによって、計算されたプロセス特性（または集めら

40

50

れたデータ)を送る。第2のチューニング要素は一つ以上の記憶された組(ファジイ論理規則組、ニューラルネットワーク構成もしくは規則組または他の組のアルゴリズムのような)を用いて、第1のチューニング要素により生じさせられたプロセス特性に基づいて制御要素のために新しいパラメータを決定する。第2のチューニング要素は次いで通信ネットワークによって制御要素に新しいチューニングパラメータを送って、制御要素または制御要素が位置している制御ループを再チューニングする。

【0010】

ここで述べられる自動チューナの実施の形態は、正確にチューニングパラメータのタイミングを制御し、必要なだけのデータを取りこんで所望のプロセス特性を決定する。なぜなら制御およびデータキャプチャ機能は、制御要素と同じ装置内で実行され、その結果、チューニング手続きの動的データキャプチャ段階の間にバスによって制御要素と通信する必要はないからである。しかしながら、この自動チューナは、制御要素内で不必要なオーバーヘッドを生じさせない。なぜなら、チューニングパラメータの計算は、制御要素から離れて位置する別の装置内で行なわれ、その結果、制御要素は、これらの機能を組み込むのにコンパイルされる必要はない。さらに、単一のチューニングパラメータ計算ルーチンは、ネットワーク内で様々な種類および様々なものの制御要素のチューニングパラメータを計算するのに用いられ得る。なぜなら、このルーチンは、チューニングされている様々な種類の制御要素の各々に対して異なる組の規則を記憶し、用いるからである。さらに、同じ自動チューナ機能ブロックまたは機能性は、制御要素を有する装置の各々内に配置可能である。なぜなら、様々な制御ループが、例えば、PI、PIDおよびファジイ論理制御ブロックを含め、様々な種類の制御ブロックを用いるとしても、これらの制御ループと関連するプロセス特性は同じ方法で決定できるからである。これらの利点によって、自動チューナをより汎用性があるようにし、制御要素を有する各装置において別個の完全な自動チューナを配置する必要がなくなる。さらに、この自動チューナは、拡張機能としてプロセス制御ネットワークの一部内に容易に実現可能である。

【0011】

【発明の実施の形態】

さて、図1を参照して、プロセス制御ネットワーク10はブロック図の形で示される。プロセス制御ネットワーク10は、ひとつ以上の集中化されたプロセスコントローラ12を含み(そのひとつのみが図1に示される)。このコントローラ12は、記憶されているプロセス制御ルーチンを実行すること、および/または、プロセス制御ネットワーク10中に分散されたフィールド装置内に位置した機能ブロックのような制御要素と通信することが出来る。コントローラ12は、例えば、フィッシャローズモントシステムズ(Fisher-Rosemont Systems)により販売されているDeltav(登録商標)コントローラであり、ハブ16およびイーサネット(登録商標)などのネットワーク18によってパソコン(PC)14のような無数のワークステーションに接続されて良い。この構成においては、PC14は、一人以上のオペレータまたはユーザにより用いられて、プロセスコントローラ12またはフィールド装置と通信し、それによりプロセス制御ネットワーク10の要素に関する情報を得、プロセス制御ネットワーク10内の要素の状態を見直すか変更し、プロセス制御ネットワーク10内の個々のフィールド装置に関する情報を得るなどを行なう。コントローラ12がDeltavコントローラであるならば、コントローラ12内のプロセス制御ルーチンの図形的表示をPC14のひとつによってユーザに与えて良く、これは、プロセス制御ルーチン内の機能ブロックまたは他の制御要素と、プロセスの制御を与えるようにこれらの機能ブロックが互いに接続される方法を示す。さらに、所望であれば、ユーザまたはオペレータは、PC14のひとつから機能ブロックまたは制御ループのうちのひとつ以上のチューニングを開始することが出来る。

【0012】

図1に示されるように、集中化コントローラ12は、プロセス(参照番号19で包括的に示される)中に位置した無数のフィールド装置に接続される。集中化コントローラ12は、標準的なタイプのI/Oカード20、22を介して、典型的なフィールド装置26、2

10

20

30

40

50

8、30、32、34、36と通信してよく、これらのフィールド装置はコントローラ12から集中化された制御を受ける。I/Oカード20は、例えば、アナログI/Oカードであってよく、これはコントローラ12をアナログフィールド装置26、28に接続し、このフィールド装置26、28は4-20mAバス38によって通信を行なう。同様に、I/Oカード22は、デジタルかまたは組み合わされたデジタルおよびアナログのI/Oカードであってよく、これは例えば4-20mAアナログフォーマットまたは他の所望のフォーマットを用いて、デジタルかまたは混合のデジタルおよびアナログのフィールド装置と通信する。当然、フィールド装置26、28、30、32、34、36は、送信機、センサ、弁ポジショナ、弁コントローラなどを含むどんな種類のフィールド装置であってよい。図1に示される例示のプロセス制御ネットワーク10のために理解されるように、フィールド装置26-36は、コントローラ12内で記憶される制御ルーチンによる集中化された制御を受けるプロセス19の部分と関連する。

10

【0013】

コントローラ12はインターフェースカード40にも通信結合され、このカード40は、プロセス制御が分散的に行なわれるプロセス制御ネットワークに（またはその一部）に接続される。図1に示される実施の形態においては、プロセス19の非集中化されたプロセス制御部分は、インターフェースカード40と、バス42と、バス42に接続された無数のフィールド装置43、44、46、48、50とを含む。装置43は、例えば、何らかのプロセス変数を測定する送信機であってよく、装置44は、プロセス19内で流量を制御するポジショナ／弁装置であってよい。図1の分散プロセス制御ネットワークは、例えば、フィールド通信プロトコルを用いるフィールドバスネットワークであってよく、インターフェースカード40は、フィールドバス通信プロトコルと関連するリンクアクティブスケジューラ（link active scheduler）であってよい。

20

【0014】

コントローラ12内に位置する集中化されたプロセス制御ルーチンは、フィールド装置26-36から、そして可能性としては43-50から、入力を受け取り、計算および制御ルーチンと関連する他の活動を行ない、ついで、I/Oカード20、22、およびインターフェースカード40によってフィールド装置にコマンドを送り、プロセス19の所望の制御を実行する。しかしながら、プロセス制御ネットワーク10の非集中化されたプロセス制御部分（すなわち図1のバス42と関連するもの）は、コントローラ12により行なわれている制御と関連して（またはその代わりに）非集中的に、それ自身のプロセス制御ルーチンを実行して良いことに注目されたい。このように、コントローラ12は、バス42に接続された装置43-50とつながり（interface）し、それらに対し制御を行なうて良いが、これらの装置は、コントローラ12により行なわれる制御と関連しないがそのかわりにバス42に接続された装置中に分散された制御ブロックまたは機能ブロックを実行してもよい。いづれにせよ、本発明の自動チューナは、図1のフィールド装置26-36および43-50またはコントローラ12のいずれかにおいて位置したいづれかの制御要素（機能ブロックのような）をチューニングするのに用いられてよい。

30

【0015】

好ましい実施の形態において、プロセス制御ネットワーク10の非集中化された部分はフィールドバス通信および制御プロトコルを用いているが、将来において開発されるプロトコルを含め、同様に他の公知のまたは所望のプロトコルを用いることが出来る。一般的に、フィールドバスプロトコルは、例えば、工場またはプラントの器械またはプロセス制御環境に位置するセンサ、アクチュエータ、装置コントローラ、弁などのような2線式ループまたはバス相互接続「フィールド」機器に、標準化された物理的インターフェースを与える全デジタル、シリアル、双方向の通信プロトコルである。事実上、フィールドバスプロトコルは、プロセス内のフィールド計器（フィールド装置）のためのローカルエリアネットワークを与え、これにより、これらのフィールド装置が、プロセス機構中に分散された位置で制御機能を実行し、全体の制御戦略を実行するためにこれらの制御機能を実行した前後に互いに通信できるようになる。フィールドバスプロトコルによって、制御機能

40

50

がプロセス制御ネットワーク中に分散可能となるので、集中化プロセスコントローラ 12 の作業負荷を低減したり、それらのフィールド装置またはプロセスのエリアのために集中化コントローラ 12 を必要としなくなる。フィールドバスプロトコルは、プロセス制御ネットワークで用いられるように開発された比較的新しい全デジタル通信プロトコルであるが、このプロトコルは当該技術で公知であり、とりわけ、テキサス州、オースチンに本社のある営利団体ではないフィールドバスファウンデーション (Fieldbus Foundation) により出版され、流通され、そこから入手できる多くの記事、パンフレットおよび仕様書に詳細に記載されている。従って、フィールドバス通信プロトコルの詳細はここでは述べられない。

【0016】

図 2 はある制御ループを構成する相互接続された機能ブロックの概略的なブロック図であり、この制御ループは、制御ループ内の機能ブロックが、バスのような通信ネットワークにより相互接続された様々な装置内に位置した、例えば、フィールドバスネットワークまたは他の分散化制御環境内で実現されてよい。図 2 に示される制御ループは、アナログ入力 (A I) 機能ブロック 60 (これは、例えば、図 1 の送信機 43 内に位置してよい) と、P I D 機能ブロック 62 と、アナログ出力 (A O) 機能ブロック 64 とを含む。図 2 の例示のループのために、P I D 機能ブロック 62 および A O 機能ブロック 64 は、図 1 のポジション / 弁 44 内に位置する。機能ブロック 60、62、64 の間の通信接続は、これらの機能ブロックのプロセスおよび制御入力と出力とを接続する線を用いて図 2 に示される。このように、A I 機能ブロック 60 の出力は、プロセス可変性測定値またはプロセスパラメータ信号を含んでよく、フィールドバス電線 42 によって P I D 機能ブロック 62 の入力に通信結合され、機能ブロック 62 は、A O 機能ブロック 64 の入力に通信結合された制御信号を含む出力を有する。A O 機能ブロック 64 の出力は、例えば弁 44 の位置を示すフィードバック信号を含み、P I D 機能ブロック 62 の制御入力に接続される。P I D 機能ブロック 62 はこのフィードバック信号を A I 機能ブロック 60 からのプロセス測定値信号 (そして可能性として他の信号または設定点) と一緒に用いて A O 機能ブロック 64 の所望の制御を行ない、それにより、A I 機能ブロック 60 により測定されたプロセス変数を制御する。図 2 の制御ループ図内の線により示される接続は、A O 機能ブロック 64 および P I D 機能ブロック 62 の場合と同様に、機能ブロックが同じフィールド装置内にあるときに、フィールド装置内で内的に行なわれてよい。

【0017】

制御要素 62 をチューニングするのに用いられる自動チューナは、ポジション / 弁 44 内に位置した自動チューニング機能ブロック 66 を含む。自動チューニング機能ブロック 66 は、チューニング手続きの動的データキャプチャ段階の動作の間に P I D 機能ブロック 62 に通信結合されて、その入力に信号を送り、出力を変更することなどにより、P I D 機能ブロック 62 を制御する。特に、チューニング手続きの動的データキャプチャ段階の間に、制御要素コントローラ 67 (これは、ハードウェアまたはソフトウェアにおいて実現される信号発生器でもよい) は、機能ブロック 60、62、64 を含む制御ループを駆動して、制御された誘導された発振手続き内に入らせる (または制御ループを駆動して他の所望のチューニング手続きを実行する)。データ収集ユニット (データ C O L) 68 は、この手続きの間に P I D 機能ブロック 62 (またはループの他の部分) により生成されまたはそれに送られるデータを集め、そして好ましくはこのデータを装置 44 のメモリ内に記憶する。プロセス特性決定ユニット (プロセス C H R) 69 は次いで、公知のまたは所望の手続きを用いて、集められたデータから、所望のプロセス特性 (プロセス、ループまたは装置特性であってよい) を決定して良い。その後、通信ユニット 70 は、計算されたプロセス特性 (またはこのような特性を決定するのに必要なデータ) をチューニングコントローラ 71 に与え、このコントローラ 71 は好ましくは、オペレータワークステーションまたは P C 14 のひとつに位置しているが、自動チューニング機能ブロック 66 が位置している装置 44 から離れて、集中化コントローラ 12 内などに位置していてもよい。

【0018】

チューニング手続きの間、自動チューニング機能ブロック 66 は、プロセスループに強制的にある変更を行なわせ、プロセス制御ループ内に存在する様々な数量または信号を感じて、プロセスまたは制御ループのひとつ以上のプロセス特性を決定する。プロセス特性（またはチューニング手続きの間に取りこまれたデータ）は、チューニングコントローラ 71 に送られるが、このコントローラ 71 はユーザインターフェース支援および一組以上の制御またはチューニングパラメータを生じさせるのに用いられるように異なる装置に位置している。PID 機能ブロック 62 を用いることはあくまで例示的であり、自動チューニング機能ブロック 66 を、なんらかの他の制御ループ内の他の種類の機能ブロックとともに用いることが出来ることが理解されるだろう。さらに、チューニングコントローラ 71 は、例えばユーザまたはオペレータインターフェース装置（ワークステーションのよ

10

【0019】

自動チューニング機能ブロック 66 の個々の要素が所望の方法で実行されると理解されるだろう。例えば、好ましい実施の形態において、これらの要素の各々は、自動チューニング機能ブロック 66 が記憶されている装置と関連するプロセッサ上で実行されるソフトウェアで実現される。この場合、制御要素コントローラ 67 は、PID 機能ブロック 62 またはチューニングされている他の機能ブロックへの入力およびその出力と通信し、それらを制御するのに用いられるべき一連の命令であってよい。同様に、データ収集ユニット 68 は、例えば、PID 機能ブロック 62 と関連するある所望の信号を監視し、装置 44 のメモリ内にこれらの信号を示すデータを記憶する。プロセス特性決定ユニット 69 は、記憶されるかまたは集められるデータからひとつ以上のプロセス特性を計算するかまたはそうでなければ決定するようにソフトウェアで実現されて良く、通信ユニット 70 は、装置 44 の通信プロトコルまたはセットアップを用いて、チューニングコントローラ 71 と通信してもよい。例えば、フィールド装置においては、通信ユニット 70 は、装置 44 内ですでに与えられている通信システムがフィールドバス 42 によってチューニングコントローラ 71 にプロセス特性を伝えるようにしてよい。フィールドバス環境においては、個々のユニット 67、68、69、70 は、独立型フィールドバス機能ブロックとして一緒に実装されても、PID 機能ブロック 62 内で与えられても良い。代替的には、ユニット 67、68、69 および／または 70 のいずれかまたは全ては、プロセス制御環境において

20

30

はソフトウェア、ハードウェアまたはファームウェアで実現可能である。自動チューニング機能ブロック 66 そして、より特定のには、チューニング手続きの間に開放ループまたは閉ループチューニング技術のような所望のチューニング手続きを用いてよい。同様に、自動チューニング機能ブロック 66 は、例えば、AO 機能ブロック 64 の出力、PID 機能ブロック 62 の設定ポイント、AI 機能ブロック 60 の出力などを含め、所望の変数を測定し、所望のプロセスまたはループ特性を決定してよい。好ましい実施の形態において、自動チューニング機能ブロック 66 は、制御され誘導された発振手続きを用いて所望のプロセス特性を決定するように構成されてよい。このような構成では、自動チューニング機能ブロック 66 のコントローラ 67 は、誘導された発振が開始される前に、PID 機能ブロック 62 の制御入力に、PID 機能ブロック 62 に送られる制御信号の値辺りに中心を置かれた $2d$ の選択可能なピーク対ピーク値を有する矩形波信号を与える。このような矩形波信号に応答して、図 2 の制御ループは、誘導された発振を行ない、AI 機能ブロック 60 の出力（すなはちプロセス変数）はピーク対ピーク振幅 $2a$ および期間 T_u を有して発振する。プロセス変数の振幅 a および矩形波信号の振幅から、プロセスの極限 (ultimate) 利得 K_u は等式 $K_u = 4d / \pi a$ に従って導出される。

40

【0020】

この場合、プロセスの極限期間 T_u は、プロセス変数の発振の期間に等しいものとして計算可能である。自動チューニング機能ブロック 66 は当然のことながら、他のデータを測定してよく、例えば、時間遅延 T_d およびプロセスの支配的時定数を含む他の所望のプロセス特性を決定してよい。

50

【0021】

このようなプロセス特性を計算した後、自動チューニング機能ブロック66は、これらの特性をチューニングコントローラ71に与え、これは、数量 K_u および T_u から、例えば、PID制御機能ブロック62の比例利得 K_p 、積分時定数 T_i 、および微分時定数 T_d を含む仮の一組の制御またはチューニングパラメータを決定する。しかしながら、所望であれば、自動チューニング機能ブロック66は、必要なデータを取りこみ、そのデータを記憶し、非実時間でバスによって記憶されたデータをチューニングコントローラ71に与え、その場合、チューニングコントローラ71は所望の方法で所望のプロセスまたはループ特性を計算して良い。

【0022】

上記の説明にかかわらず、自動チューニング機能ブロック66はPI、PID、ファジイ論理または他の種類の制御要素を有する制御ループに対し公知の無数のルーチンを含む所望の動的データキャプチャルーチンを実行してよい。例えば、自動チューニング機能ブロック66は米国特許第4,549,123号（その開示をここに引用により援用する）に記載されているハグラントーアストロム社のリレーチューニング手続きのような制御された誘導された発振手続きを実行してよい。リレータイプのチューニング方法では、自動チューニング機能ブロック66はプロセスループ（機能ブロック60、62、64で構成された）を、例えば非線形フィードバック機能を用いて自己発振させ、プロセスループの応答を測定して、極限利得および極限期間のような所望のプロセス特性を決定する。

【0023】

代替的には、自動チューニング機能ブロック66は、1993年8月11日に出願され、「自動チューニングを備えたファジイ論理制御のための方法および装置 "Method and Apparatus for Fuzzy logic Control with Automatic Tuning"」と題された同時係属中の親出願連続番号08/105,899（米国特許番号6,330,480。その開示をここに引用により援用する）に記載されているチューナのいずれかの設計を組み込んで良い。例えば、自動チューニング機能ブロック66（特にプロセス特性計算機69）は、モデル突き合わせチューニング手続きを行なって良く、この手続きにおいては、プロセス変数に対しシグネチャ(signature)解析を行なって、複数の記憶された数学的モデルから、プロセスまたはループをもっとも正確に特徴付けるモデルを選択する。プロセスループのプロセス特性はついで、選択されたモデルから決定される。

【0024】

さらに、自動チューニング機能ブロック66は、米国特許第4602,326号（その開示をここに引用により援用される）において開示されるもののようなプロセス特性化のパターン認識方法を用いてループのプロセス特性を決定してよい。チューニングのパターン認識方法において、プロセスまたはループの特性は、それがプロセス混乱状況に反応するときにプロセス変数を観察することにより決定される。プロセス混乱状況の結果として生成されたプロセス変数のパターンは次いで解析されて、プロセスまたはループの所望の特性を決定する。

【0025】

さらに、自動チューニング機能ブロック66は、例えば、米国特許第5,283,729号、および1993年5月28日に出願され、「プロセスコントローラを自動的にチューニングするシステムおよび方法 "System and Method For Automatically Tuning a Process Controller"」と題された同時係属中の米国特許出願連続番号08/070,090（米国特許番号5,453,925。その開示をここに引用により援用する）に開示されたもののようなプロセス（例えばループ）の特性を決定する信号注入方法を用いて良い。しかしながら、所望のごとく、他のどんなチューニング手続きが用いられても良く、自動チューニングコントローラ67は、ここに述べられたまたは引用された具体的なチューニング手続きとともに用いることに限定されない。

【0026】

同様に、チューニングコントローラ71は、自動チューニング機能ブロック66により計

10

20

30

40

50

算されるか集められたプロセスまたはループ特性（または生データ）からチューニングパラメータを決定する所望の方法を用いて良い。例えば、チューニングコントローラ71は、米国特許第5、748、467号（その開示をここに引用により援用する）に記載され引用されるチューニングパラメータ計算方法、例えば、いくつか例を挙げると、内部モデル制御（IMC）チューニング、コーエン（Cohen）およびクーン（Coon）チューニング、ツィーグラ・ニコルズチューニング（Ziegler-Nichols）、または修正されたツィーグラ・ニコルズチューニングのいくつかを用いてよい。

一般に、チューニングコントローラ71は、ひと組以上の規則（ファジィ論理規則組、ニューラルネットワーク機器構成または規則組、または線形か非線形アルゴリズムの組のような）を記憶して、これらを用いて、チューニングされている機能ブロックの種類に従って利得および時定数のようなチューニングパラメータを決定する。さらに、チューニングコントローラ71（これは、様々な装置内のどんな数の様々な自動チューニング機能ブロックと通信してよい）は、様々な種類の制御ブロックまたはループのため異なる組のチューニングパラメータ計算規則を記憶して良く、そしてこのようにして、プロセス制御システム内のPID、PI、ファジィ論理または他の制御要素のいずれかか全てをチューニングするのに用いられてよい。チューニングコントローラ71は、様々な種類の制御要素とともに用いられるべき多くの異なる組の規則を記憶して良く、自動チューニング機能ブロック66により決定される動的プロセス特性に基づいて新しいチューニングパラメータを計算するときにどんな組の規則であれ適切なものを応用することができるので、単一のチューニングコントローラ71だけがプロセス制御ネットワーク内に位置する必要がある。チューニングコントローラ71が新しいチューニングパラメータを決定した後、それは、バス42によって図2のPID機能ブロック62のような適切な制御要素にこれらのパラメータを与える。所望であれば、チューニングコントローラ71は、チューニングパラメータを自動チューニング機能ブロック66に与えて良く、自動チューニング機能ブロック66は次いでこれらのパラメータをPID機能ブロック62に与えて良い。

チューニングの動的データキャプチャ段階の間に実行される機能がチューニングの規則アプリケーション段階の間に実行される機能から分離されるような、上で述べた分離された機能セットアップを用いて、自動チューナは、チューニング手続きの動的データキャプチャ段階の間に非常に正確に制御ループの動作のタイミングを制御することができる。さらに、この自動チューナは、チューニングされている制御ループのプロセス特性を正確に決定するのに必要なだけのデータを検出し、記憶することが出来る。なぜなら、データ通信は、バスによって実時間において生じていないからである。また、プロセスまたはループ特性（かなり複雑である）に基づいて新しいチューニングパラメータを決定することと関連する規則適用機能は、装置においてまたはループを制御するのに用いられる機能ブロックにより実行されないもので、これらの機能は、動作ループ内に符号化される必要はない。これにより、全体のチューニングルーチンが、チューニングされている制御ブロック内に位置するシステムと比較すると、ループのオーバーヘッドを低減する。

【0027】

さて図3を参照して、本発明の自動チューナのさらなる実施の形態は、集中化および非集中化された方法で実行される制御ブロックまたは制御機能を有する環境内で用いられるように示される。この例においては、AI機能ブロック60およびAO機能ブロック64は図1の集中化コントローラ12内に位置しそれにより実行され、コントローラ12は、送信機装置および／ポシヨナ装置（図1の装置26、28、30、32、34、または36のいずれであってもよい。）と通信して、これらの機能ブロックの入力および出力を得る。しかしながら、PID機能ブロック62は図1の非集中化フィールドバスネットワーク内のポシヨナ／弁装置44内に記憶され、これにより実行され、図1のフィールドバスリンク42およびインターフェースカード40によって、集中化されたコントローラ12と通信する。

【0028】

この実施の形態においては、集中化コントローラ12は、影（shadow）PID機能ブロッ

10

20

30

40

50

ク 7 2 を含み（破線ブロックにより示される）、これは、P I D 機能ブロック 6 2 と関連する入力および出力ならびに他のデータを映し、P I D 機能ブロック 6 2 がまるで集中化コントローラ 1 2 内に記憶されそれにより実行されているかのように A I 機能ブロック 6 0 および A O 機能ブロック 6 4 と通信する。より特定的には、影 P I D 機能ブロック 7 2 が A I 機能ブロック 6 0 および A O 機能ブロック 6 4 からリンクによって、入力を受け取ると、それはすぐにポジショナ／弁装置 4 4 内の実際の P I D 機能ブロック 6 2 にこれらの入力を送る。図 3 に破線で示されるように、影 P I D 機能ブロック 7 2 および実際の P I D 機能ブロック 6 2 間の通信は、フィールド装置 4 4 と関連する通信プロトコルを用いて、この場合は、フィールドバスプロトコルを用いて行なわれる。実際の P I D 機能ブロック 6 2 は次いで、A I 機能ブロック 6 0 および A O 機能ブロック 6 4 がフィールドバスネットワーク内のフィールドバス機能ブロックであるかのように、標準的な方法で、受け取られた入力に基づいてその制御機能を実行する。

10

【0029】

同様に、影 P I D 機能ブロック 7 2 は、実際の P I D 機能ブロック 6 2 の状態を示す出力および他のデータの周期的に更新したものを受信して、適切な時間に A I 機能ブロック 6 0 および A O 機能ブロック 6 4 にコントローラ 1 2 内のこのようなデータをリンクによって伝えるように構成されている。このようにして、A I 機能ブロック 6 0 および A O 機能ブロック 6 4 には、P I D 制御機能が集中化コントローラ 1 2 内で与えられているように見える。これらのブロックは、これらのブロックが集中化コントローラ 1 2 により実行される他の機能ブロックと通信するのと同じ方法で影 P I D 機能ブロック 7 2 と通信可能である。影機能ブロックの実現化例の詳細は、本発明の主題ではないが、1998 年 9 月 10 日に出願された「プロセス制御ネットワークで用いられる影機能ブロックインターフェース “A Shadow Function Block Interface For Use in a Process Control Network”」と題された米国出願連続番号 09 / 151、084（米国特許番号 6,738,388）に詳細に説明され、これは本発明の譲受人により譲渡され、その開示はここに引用により援用される。

20

【0030】

図 3 の実施の形態においては、自動チューニング機能ブロック 6 6 はポジショナ／弁 4 4 内に位置し、図 2 に関して説明されたように、チューニング手続きを実行するように実際の P I D 機能ブロック 6 2 と通信し、これを制御する。P I D 機能ブロック 6 2 と同様に、自動チューニング機能ブロック 6 6 はコントローラ 1 2 内に位置する影自動チューニング機能ブロック 7 6 を有する。実際の自動チューニング機能ブロック 6 6 は、影自動チューニング機能ブロック 7 6 と通信して、データ（計算されたプロセスまたはループ特性のような）を影自動チューニング機能ブロック 7 6 に送り、自動チューニング機能ブロック 7 6 からコマンドまたは他のデータを受信する。図 2 の実施の形態と同様に、自動チューニングのチューニングコントローラ 7 1 はワークステーションまたは P C 1 4 内に記憶されこれにより実行されるが、その代わりに、コントローラ 1 2 内または他の所望の装置内であってよい。チューニングコントローラ 7 1 は影自動チューニング機能ブロック 7 6 と通信して、例えば、チューニングを開始するか、自動チューニング機能ブロック 6 6 により生じさせられたプロセスまたはループ特性を受け取るか、または自動チューニング機能ブロック 6 6 により集められたまたはその状態を示すデータを受信する。

30

40

【0031】

チューニングが図 3 の実施の形態において望まれるならば、チューニングコントローラ 7 1 はコマンドを（例えば、ユーザからまたはプロセス制御ネットワーク内のさらなるコントローラから）受け取り、チューニングを開始する。チューニングコントローラ 7 1 は次いでチューニング開始コマンドを影自動チューニング機能ブロック 7 6 に送り、ブロック 7 6 はこのコマンドを実際の自動チューニング機能ブロック 6 6 に送る。このコマンドに応答して、実際の自動チューニング機能ブロック 6 6 は、図 2 の実施の形態に関して上で述べたようなもののいくつかのようなチューニング手続きを開始し、チューニング手続きの間にループにより生成されたデータを集め、次いでひとつ以上の所望のプロセス特性を

50

計算するかまたは決定する。実際の自動チューニング機能ブロック66は、次いで、これらの特性を（または集められたデータ）を影自動チューニング機能ブロック76に与え、ブロック76は、この情報をチューニングコントローラ71に伝える。

【0032】

代替的には、影自動チューニング機能ブロック76またはチューニングコントローラ71は、影PID機能ブロック72の状態を変更してよい（影PID機能ブロック72は実際のPID機能ブロック62の状態を変更するが、これは、実際のPID機能ブロック62に強制的に実際の自動機能ブロック66を呼び出させるかまたはこれと通信させて、それによってチューニング手続きを実行させる）。この場合、PID機能ブロック62の状態に変化があるとすぐに、影自動チューニング機能ブロック76は開始されるかまたは自動的にインストールされてよい。

10

【0033】

いずれにせよ、チューニングコントローラ71は、所望のチューニングパラメータ計算方法を用いて、受信されたデータ（計算されたループまたはプロセス特性のような）に基づいて、新しいチューニングパラメータを決定する。チューニングコントローラ71は次いでこのようなチューニングパラメータを影PID機能ブロック72（または必要に応じて他のどんな機能ブロックにでも）に送り、影PID機能ブロック72は、これらの新しいチューニングパラメータを実際のPID機能ブロック62を送り、ここでそれらはループの正規動作の間にPID機能ブロック62により用いられる。

20

【0034】

影自動チューニング機能ブロック76を用いることは望ましいが必要ではなく、その代わりに、チューニングコントローラ71は、直接自動チューニング機能ブロック66と通信してチューニング手続きを実行することは理解されるであろう。さらに、図2および3の実施の形態は、PID機能ブロック62とは別個の機能ブロックとして自動チューニング機能ブロック66を示しているが、そのかわりに自動チューニング機能ブロック66の機能性は、別個の機能ブロックであるかわりにPID（または他の制御）機能ブロック62内に組み込み可能である。

【0035】

好ましくは、自動チューニング影機能ブロック76は、フィールドバス装置の自動チューニング機能ブロック66の属性と同一の属性を有する。しかしながら、ある実施の形態においては、STRATEGY、ALERT_KEY、MODE_BLK、およびBLOCK_ERRの標準的なフィールドバス属性は、実際の自動チューニング機能ブロック66において与えられるが、影自動チューニング機能ブロック76において与えられる必要はない。また、自動チューニング影機能ブロック76はユーザに直接見える必要はない。なぜなら、それは、チューニング手続きの識別段階（すなはち動的データキャプチャ段階）の間のみ、制御を開始し、チューニングコントローラ71と自動チューニング機能ブロック66との間の通信を与えるのに用いられるからである。にもかかわらず、自動チューニング影機能ブロック76のパラメータは、チューニングコントローラ71およびチューニングユーザインターフェースアプリケーションにより利用可能であり、これらにより用いられ、そして所望であれば、ユーザアプリケーションにより得られる。さらに、フィールドバスPID機能ブロック62は好ましくは、自動チューニング機能ブロック66の開始指数を含むように修正され、この指数は、影自動チューニング機能ブロック76により用いられるように利用可能である。これによって、PID機能ブロック62は自動チューニング機能ブロック66により制御可能となる。

30

40

【0036】

プロセス特性情報は、例えば、影自動チューニング機能ブロック76だけでなくフィールドバス自動チューニング機能ブロック66内に、他の種類のフィールドバス機能ブロックと関連する他の属性と同じ方法で属性として含まれる。当業者は、用いられている様々な種類のチューニング、および自動チューナが位置するシステムの様々な局面に依って、様々な属性を影自動チューニング機能ブロック76だけでなく実際の自動チューニング機能

50

ブロック 66 において与えることが可能である。自動チューニング機能ブロックにより与えられる例示のプロセス特性は以下の通りに規定される：T__I P G A I N-積分プロセス利得-入力におけるユニットステップ変更に対するプロセス出力の変更のレート；T__U G A I N-極限利得-ループがかろうじて安定している、つまり、安定性と不安定性との間の境界上にある比例コントローラ利得の値；T__U P E R I O D-極限期間-そのコントローラ利得がプロセスの極限利得に設定されるループの発振期間；T__P S G A I N-プロセス静的利得-プロセス出力変更のプロセス入力変更に対する比；T__P T I M E C；プロセス時定数-システムがステップ入力にตอบสนองし始めるときと、システムがその最終の着実な状態の値の 63 パーセントに達するときとの間の持続時間；および T__P D T I M E-プロセス遅延時間-制御動作が行なわれているときとプロセスがこの動作にตอบสนองするときとの間の時間の長さ。

10

【0037】

チューニングテストが完了したとき、これらの属性は、自動チューニング機能ブロック 66 において計算され、次いで、例えば、ワークステーション 14 内にチューニングコントローラ 71 に提示され、これは、これらのプロセス特性情報または属性のうちのひとつ以上を用いて、例えば、P I または P I D またはファジイ論理コントローラのための利得、積分時間（リセットの逆）および微分時間（レート）のようなチューニングパラメータを決定する。ある実施の形態においては、上に挙げられた属性は、浮動小数点データであってよく、静的データとして不揮発性メモリ内に記憶されて良く、リードオンリーデータとしてアクセスされてよい。また、上に挙げられた属性の各々は、フィールドバスまたは D e l t a V プロトコルのビュー 1 および 3 において利用可能であってよい。フィールドバスまたは D e l t a V プロトコルにおけるビューリストは、機能ブロックからの情報に対する単一の要求（すなはちあるビュー）により与えられるパラメータを規定すると理解されるだろう。このように、ビュー要求を用いて、影自動チューニング機能ブロック 66 は、自動チューニング機能ブロック 66 から大多数の利用可能な属性の最新値を得て良い。

20

【0038】

さらに、上に挙げられた属性またはプロセス特性は自動チューナのひとつの可能な実現化例と関連し、本発明に従って動作するように他の方法で構成される他の自動チューナにより与えられる他の属性および情報があることが理解されるだろう。

【0039】

このような他のプロセス特性情報は、極限利得、静的利得、および積分利得のような所望の種類の利得と、極限期間、不動作時間、および微分時間のような時間と、プロセスと関連する時定数と、他の所望の情報とを含んで良い。

30

【0040】

理解されるように、実行される自動チューニングの正確な種類は本発明にとっては重要でなく、上で挙げられたリストは、チューニングコントローラ 71 に利用可能にされてチューニングコントローラ 71 がひとつ以上のチューニングパラメータを決定することが可能なプロセス特性情報のなんらかの種類の単なる例にすぎない。所望であれば、チューニングコントローラ 71 は、なんらかの種類の P I D、ファジイ論理、ニューラルネットワークなどのプロセスを用いて、極限利得、極限期間、および不動作時間のような入力から、コントローラ利得、リセットなどのようなチューニングパラメータを決定して良い。

40

【0041】

好ましい実施の形態においては、自動チューニングアプリケーション（すなはちチューニングコントローラ 71）は、フィールドバス装置（すなはち実際の P I D 機能ブロック 62）内で関連する機能ブロックに自動チューニングが行なわれるべきであると示すため、影 P I D 機能ブロック 72 の T U N S K I 属性に書き込みをする。この属性変更にตอบสนองして、影自動チューニングブロック 76 のインスタンスが集中化コントローラ 12 内で作成される。（付随的に、チューニングコントローラ 71 は、いつ自動チューニングが完了するかを示すために T U N S K I 属性に書き込みをし、これによって、影自動チューニング機能 76 は廃棄される）。影自動チューニング機能ブロック 76 が作成された後、フィー

50

ルドバス装置内の関連する自動チューニング機能ブロック 6 6 の状態が読まれる。自動チューニング機能ブロック 6 6 の状態がクリア ($T_Request = 0$) であるならば、チューニングは着手されてよい。

【0042】

しかしながら、もし、 $T_Request$ 属性が 0 でなければ、チューニングはフィールド装置内でアクティブ (活動最中) である。この場合、チューニング要求は尊重されず、影自動チューニング機能ブロック 7 6 のインスタンスは消去され、それにより自動チューニングを停止する。

【0043】

影自動チューニング機能ブロック 7 6 のインスタンスが作成されると、その静的および動的属性は、自動チューニング機能ブロック 6 6 のものに反映する。これらの値は、まるで集中化コントローラ 1 2 内の PID 機能ブロックがチューニングされているかのように自動チューニングアプリケーションによるアクセスされるだろう。このように、もし自動チューニングアプリケーション (コントローラ) 7 1 が新しい値を影自動チューニング機能ブロック 7 6 の属性に書き込むならば、影自動チューニング機能ブロック 7 6 は、関連の自動チューニング機能ブロック 6 6 に対しこの属性値の書き込みを開始する。同様に、自動チューニング機能 6 6 がフィールドバス装置内でアクティブに働いているとき、影自動チューニング機能ブロック 7 6 は周期的に自動チューニング機能ブロック 6 6 の動的属性値を得る (フィールドバスプロトコルにおけるビュー 3)。自動チューニング機能ブロック 6 6 が機能ブロックとして規定されているので、このビューは他のブロックと同じ方法でアクセスされる。ビュー要求に応答して与えられた動的属性値は次いで、その動的属性値を更新させるために影ブロック 7 6 において用いられる。フィールドバスインターフェースカード 4 0 (図 1) が静的属性値において変更を見ると、それは、自動チューニング機能ブロック 6 6 から静的属性の全て (ビュー 4) を要求する。この要求に応答して受け取られた静的値は次いで、影自動チューニング機能ブロック 7 6 の属性を更新させるのに用いられる。

【0044】

さらに、集中化コントローラ 1 2 と関連する装置ライブラリ内の装置が PID 機能ブロックを含むとき、このテンプレート情報は、図 3 のポジショナ / 弁 4 4 内のような装置内でそれと関連する自動チューニング機能ブロックの開始のオブジェクトディクショナリ指数を規定する PID 機能ブロックと関連するひとつの静的属性を含んで良い。一般的に、ひとつ以上の PID (および / または他の制御) ブロックを含む装置内にひとつの自動チューニング機能ブロックが存在する必要がある。しかしながら、自動チューニング機能ブロック 6 6 は好ましくは、装置テンプレート内のブロックとして規定されないだろうし、したがってライブラリには示されないだろう。にもかかわらず、影自動チューニング機能ブロック 7 6 が集中化コントローラ 1 2 により作成されるとき、自動チューニング機能ブロック 6 6 の指数は影自動チューニングブロック 7 6 により用いられるだろう。この指数に基づいて、プロキシは、自動チューニング機能ブロック 6 6 のためのフィールドバスインターフェースカード 4 0 において確立されるだろう。このようにして、遠隔装置 (例えばフィールドバス) 装置における自動チューニング属性は、自動チューナアプリケーションが活性化され、関連の PID 機能ブロック 6 2 がチューニングのために選択されてしまうと、アクセスされる。PID 機能ブロック 6 2 が遠隔装置で実行されているとき、その装置内の自動チューニング機能ブロックに対する PID 機能ブロック基準がわかるだろう。もし PID 機能ブロックが集中化コントローラ 1 2 内で実行されているならば、自動チューニングブロックは、自動チューニングの開始時に PID ブロックによってインスタンス生成が行なわれるだろう。

【0045】

さて図 4 を参照して、別の実施の形態は、実際の自動チューニング機能ブロック 6 6 がコントローラ 1 2 内に位置し、コントローラ 1 2 内に位置する PID 機能ブロック 6 2 をチューニングするのに用いられる場合を示す。この構成は、例えば、非スマート装置または

10

20

30

40

50

他の装置を制御する制御ブロックをチューニングすることが望まれる場合、用いられて良く、その動作は集中化コントローラ 12 により実行される制御機能ブロックにより制御される。チューニングコントローラ 71 は、コントローラ 71 が実際の自動チューニング機能ブロック 66 と直接通信し、機能ブロック 66 が実際の P I D 機能ブロック 62 と通信し、その両方がコントローラ 12 内に記憶されることを除いては、図 3 の実施の形態と同じ方法で動作することが理解されるだろう。

【0046】

好ましい実施の形態において、自動チューニング機能ブロック 66 は、遠隔（例えばフィールドバス）装置内でまたはコントローラ 12 内で修正子（modifier）機能ブロックとして与えられる。一般的に、修正子機能ブロックは、オリジナルまたは基本機能ブロック（P I D 機能ブロック 62 のような）のアルゴリズムを修正する機能ブロックであり、オリジナルまたは基本機能ブロックの一部ではない。

10

【0047】

修正子機能ブロックの概念は図 5 に関して詳細に述べられるだろう。図 5 は、基本またはオリジナル機能ブロック 78 を示し、これは、この例では、図 2、3、および 4 の P I D 機能ブロック 62 である。理解されるように、基礎またはオリジナル機能ブロック 78 は、プロセッサにより実行されるコードまたは他の種類のコードのラインを有するアルゴリズム 80 を含む。ポイント 82 および 84 として図 5 に示されるコード内のひとつまたは多くの点では、中断または分岐点はアルゴリズム 80 内に設定される。これらの分岐点 82、84 のひとつまたは両方は、アルゴリズム 80 内で向上させられた性能を与えるのに用いられてよく、このような向上させられた性能は、アルゴリズム 94 を有する修正子機能 90 ブロックにより与えられる。

20

【0048】

一般的に言って、分岐点 82、84 は、他の所望の分岐点といっしょに、それが作成されるときに基本機能ブロック 78 内にセットアップされる。しかしながら、これらの分岐点のコンテンツは、基本機能ブロック 78 をコンパイルした後、修正子機能ブロック 90 かまたは基本アルゴリズム 80 内の次ぎのラインのコードに対するポインタを含むように変更されてよい。最初に、中断点 82 でのポインタは、基本アルゴリズム 80 内の次ぎのラインのコードを指し示すように、またはアルゴリズム 80 がこれに続くようにするために設定され、したがって、アルゴリズム 80 が、向上させられた性能なしに動作するようになる。しかしながら、なんらかの向上させられた性能または機能性が基本機能ブロック 78 に付加されることが望まれる場合、ひとつ以上の分岐点でのポインタ（例えば点 82）は、アルゴリズム 80 がその分岐点に達すると、アルゴリズム 80 が修正子機能ブロック 90 内のアルゴリズム 94 を呼び出すか、またはそうでなければ、アルゴリズム 94 に制御を転送するように設定される。所望であれば、ポインタ 82 は、メモリ内で修正子機能ブロック 90 のアドレスを示すアドレスを格納してよく、修正子機能ブロック 90 に制御を渡すのに用いられる他の分岐装置または方法であってよい。

30

【0049】

制御が修正子機能ブロック 90 に渡されると、ある予め規定されたデータが修正子機能ブロック 90 に送られるかまたはそれに利用可能にされ、修正子機能ブロック 90 は、アルゴリズム 94 の動作の間にこのデータを用いるか変更する。このデータは好ましくは、インターフェース 96 として図 5 に示される固定されたまたは予め規定されたインターフェースを介して与えられる。インターフェース 96 は典型的には、基本機能ブロック 78 内のどこに制御を戻すべきかで修正子機能ブロック 90 で用いられるのに必要なパラメータまたはデータ値（またはアドレスのようなそれに対するポインタ）のうちのいずれか、および／または、制御を取り、実行し、制御を基本機能ブロック 78 に制御を戻すために必要とされる他のデータまたは情報を含む。制御を受け取ると、修正子機能ブロック 90 は、インターフェース 96 によって与えられたデータ上に作用するそのアルゴリズム 94 を実行し、そして、終了すると、制御がそこから解放された中断点または分岐点からすぐに先に進むかまたは続くような（実行の意味で）基本機能ブロック 78 のアルゴリズム 80

40

50

内の点に制御を戻す。修正子機能ブロック 96 は、インターフェース 96 によって（修正子機能ブロック 90 に制御を送るのに用いられるものと同じであるかまたは異なるインターフェースであってよく）あるデータを戻し、そのデータは基本機能ブロック 78 内のアルゴリズム 80 により用いられる。

【0050】

2つの分岐点 82、84 が図 5 に示されているが、1つ、3つ、または他の数の分岐点が、アルゴリズム 80 内のひとつ以上の点で向上させられた性能を与えるためにオリジナル機能ブロック 80 内に配置可能である。さらに、基本機能ブロック 78 内の分岐点は、修正子機能ブロック 90 内のアルゴリズム 94 の異なる部分またはサブルーチンを呼び出すために設定されて、それにより、基本アルゴリズム 80 の異なる部分で異なる向上させられた機能を与えてよい。この機能性は、基本アルゴリズム 80 内で様々な点に付加された様々な性能を有することが必要であるかもしれない基本機能ブロック 78 内でチューニングのような完全に新しい手続きを実現するのに必要であるかもしれない。当然、基本アルゴリズム 80 内の様々な分岐点は、単一の修正子機能ブロックの同じまたは異なる部分を呼び出すためにセットアップされても、様々な修正子機能ブロックを呼び出すのにセットアップされてもよい。

【0051】

最初に販売されている場合、基本機能ブロック 78 には分岐点 82、84（他の所望の分岐点だけでなく）が設けられてよく、これにより、修正子機能ブロックの向上させられた性能を後日ユーザに与えられる。ユーザが修正子機能ブロックと関連する機能性を実現する必要がないかまたはこれを欲しないとき、ユーザは、ナル値、または、アルゴリズム 80 が修正子機能ブロックを呼び出すことなく継続すべきであると示す、基本アルゴリズム 80 内の分岐点での他の何らかの指示を記憶させる必要があるだけである。修正子機能ブロックの機能が付加されるべきであるとき、基本アルゴリズム 80 内のひとつ以上の分岐点でのポインタの値は、その分岐点に達すると、基本アルゴリズム 80 が適切な修正子機能ブロックを呼び出すように変更される。典型的には、これは、基本機能ブロック 78 を再コンパイルすることなく行なうことが可能である。その後、修正子機能ブロック 90 は、基本アルゴリズム 80 が分岐点に達すると呼び出される。呼び出されると、修正子機能ブロック 90 は、基本機能ブロック 78 内でアルゴリズム 80 の機能性を向上させるか変更させるためにアルゴリズム 94 を実行する。しかしながら、向上させられたかまたは新しい機能性がもはや望まれない場合、例えば、チューニング手続きの終わりに、分岐点は、基本機能ブロック 78 が修正子機能ブロック 90 を呼び出さないようにリセット可能である。

【0052】

このように、基本機能ブロック 78 は、更新された性能なしに最初に販売可能であり、これは、その機能ブロックと関連するドキュメンテーションおよびオーバーヘッドを減らす。しかしながら、基本機能ブロック 78 は、単に適切な修正子機能ブロック 90 を、基本機能ブロック 78 が記憶されている装置に付加し、基本機能ブロック 78 のアルゴリズム 80 内のポインタを変更するだけで更新可能である。これにより、完全な新しい基本ブロックをインストールする必要なく、また基本機能ブロック 78 を再コンパイルする必要なく、基本機能ブロックを向上させるかまたは後日変更することが出来、これによって、基本機能ブロック 78 を容易に更新できる。さらに、様々な修正子機能ブロックは、システムを更新させることに関してユーザにより多くの選択肢を与える同じ基本機能ブロックのために開発可能である。これにより、機能ブロックプロバイダが比較的簡単に修正子機能ブロックの様々なバージョンを与えたり更新したりできるようにもなる。さらに、基本機能ブロック 78 のアルゴリズム 80 は、修正子機能ブロックに対する標準的なインターフェースを支援するようにブロックが設計される付加的な性能を与えるように修正される必要はない。同様に、付加された機能性のための支援は、機器構成の間またはチューニングアプリケーションのようなカスタムアプリケーションを通じて設定されたブロック属性により示されるように呼び出されるだけである。これは、修正子機能ブロックにより与えら

れる特徴がプロセスの正規動作の間に必要でないときに、メモリおよびCPU要件を低減させる。理解されるように、修正子機能ブロックを用いるとき、基本機能ブロックは、それが修正子機能ブロックを呼び出す性能（ケイパビリティ）を有しなければならない、すなはちその中にひとつ以上のリセット可能な分岐点を有しなければならないという事実を除いては同じままである。

【0053】

フィールドバス環境において用いられるとき、修正子機能ブロック90（この場合自動チューニング機能ブロック66）は、基礎機能ブロック（例えばPID機能ブロック62）と同じ実行時間フレーム内で実行されるだろう。さらに、修正子機能ブロック90は、ユーザにはシステム内の機能ブロックとして見える必要はない。なぜならば、それは単に基本機能ブロック78のアルゴリズム80を修正しているだけであるからだ。しかしながら、所望であれば、自動チューナアプリケーションユーザインターフェースは、周知のOPCプロトコルを用いて修正子機能ブロック90から情報を得ることが出来る。さらに、単に、修正子機能ブロック90のコードを単に変更することにより基本機能ブロック78のコードを変更することは容易であるが、修正子機能ブロック90は基本機能ブロック78よりもより簡単にシステムに付加可能であり、システムから消去可能である。これは、高度の制御性能の開発を非常にスピードアップさせ、カスタマがそれらの機能を用いる特定のアプリケーションを有するときに、そのカスタマが基本機能ブロックにオーバーヘッドを付加しさえすればよいということを意味する。つまり、修正子機能ブロックに与えられる付加的な機能性が必要でないときに、カスタマは、修正子機能ブロックを用いることおよびコンパイルすることなくシステムのオーバーヘッドを低減させる。

【0054】

この説明は、単一の自動チューニング機能ブロック66と単一のチューニングコントローラ71とを有する自動チューナの実現化およびこれを用いることに向けられてきたが、自動チューナは、単一または多数のチューニングコントローラ71、および／または異なる装置に位置する多数の自動チューニング機能ブロック66を含むことが可能であることが理解されるだろう。さらに、単一の自動チューニング機能ブロック66は、同じ装置内で多数の異なる制御ブロックの動的データキャプチャ段階を実行するのに用いられることが出来、そしてそのようなわけで2つ以上の制御ブロックを有する装置は依然としてひとつの自動チューニング機能ブロックを有する必要があるだけである。同様に、同じ種類の自動チューニング機能ブロックは、異なる種類の制御ブロック（PI、PIDおよびファジイ論理制御ブロックのような）とともに用いられることが出来る。なぜならば、同じ動的データキャプチャ技術を、これらの異なる種類の制御要素の各々を有するループのために用いることが出来るからである。自動チューニング機能ブロックは、外部プロセス制御通信プロトコル（フィールドバスプロトコル）を用いて実現可能であり、フィールドバスプロトコルにより特定されかつこれにより支援される様々な機能ブロックと同様であるかまたは同じである機能ブロックを含め、なんらかの種類の機能ブロックと通信するのに用いられて良い。さらに、自動チューニング機能ブロックはここではフィールドバス「機能ブロック」として述べられているが、ここで「機能ブロック」という表現を用いることは、フィールドバスプロトコルが機能ブロックとして特定するものに限定されないが、そのかわりに、なんらかの種類の制御システムおよび／または通信プロトコルと関連し、なんらかの制御機能を実行するのに用いられるどんな他の種類のブロック、プログラム、ハードウェア、ファームウェアなどを含む。このように、機能ブロックは典型的にはオブジェクト指向プログラミング環境内でオブジェクトの形をとっているが、その必要はなく、その代わりに、プロセス制御環境内で特定の制御（入力および出力を含む）機能を実行するのに用いられる他の論理ユニットであってよい。

【0055】

ここで説明された自動チューニング機能ブロックおよびチューニングコントローラは好ましくは、コントローラまたは他のプロセス制御装置内に記憶されたソフトウェアで実行されているが、それらは、所望のごとくハードウェア、ファームウェア、などで代替的にまた

は付加的に実行されてもよい。ソフトウェアで実行されるときは、本発明の自動チューニング機能ブロックおよびチューニングコントローラは、磁気ディスク、レーザディスク、または他の記憶媒体上のようなコンピュータ読み取り可能なメモリ内で、コンピュータ、コントローラ、フィールド装置のRAMまたはROMなどにおいて記憶されてよい。同様に、このソフトウェアは、電話回線、インターネットなどのような通信チャンネルによってを含め、公知のまたは所望の送り方法によって、ユーザまたは装置に送られても良い。

【0056】

また、本発明の自動チューナは、一組のフィールドバス装置を用いて非集中化または分散化した方法でプロセス制御機能を実行するプロセス制御ネットワークと関連して詳細に説明されたが、本発明の自動チューナは、アナログおよび／またはデジタル通信を支援する2線式バスおよびプロトコル以外のものを信頼するプロトコルを含め、他の種類のフィールド装置および通信プロトコルを用いて制御機能を実行するプロセス制御ネットワークとともに用いられて良いことが理解されるだろう。例えば、本発明の自動チューナは、HART、PROFIBUSなどの通信プロトコル、または既存であるか将来において開発されるかもしれない他の通信プロトコルに従う装置を用いるプロセス制御ネットワークにおいて用いられることが出来る。

【0057】

本発明は、具体例を参照して説明されたが、これらは、あくまでも例示的であり、本発明を限定するものではないが、本発明の精神および範囲を逸脱することなく、開示された実施の形態に、変更、追加、およびまたは消去を加えてよいことは当業者には明らかである。

【図面の簡単な説明】

【図1】自動チューナを用いる分散化制御要素を有するプロセス制御ネットワークの概略的なブロック図である。

【図2】分散化プロセス制御環境内の制御ループにおいて用いられる自動チューナの実施の形態の概略的なブロック図である。

【図3】分散化および集中化制御要素の両方を有するプロセス制御環境内で制御ループにおいて用いられる自動チューナのさらなる実施の形態の概略的なブロック図である。

【図4】集中化制御要素を有するプロセス制御環境内の制御ループにおいて用いられる自動チューナのさらなる実施の形態の概略的なブロック図である。

【図5】プロセス制御環境内で修正子機能ブロックとして実行される自動チューニング機能ブロックの概略的なブロック図である。

【符号の説明】

- 44 第1の装置
- 14 第2の装置
- 62 制御要素
- 44 第1のチューニング要素
- 67 制御要素コントローラ
- 68 データ収集ユニット
- 79 コミュニケータ
- 71 第2のチューニング要素

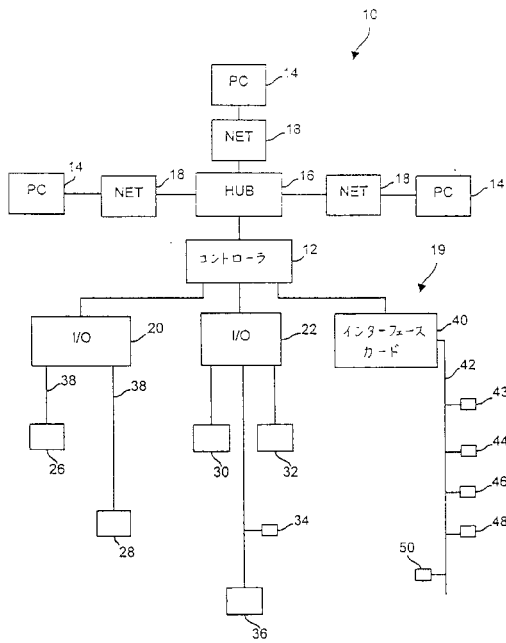
10

20

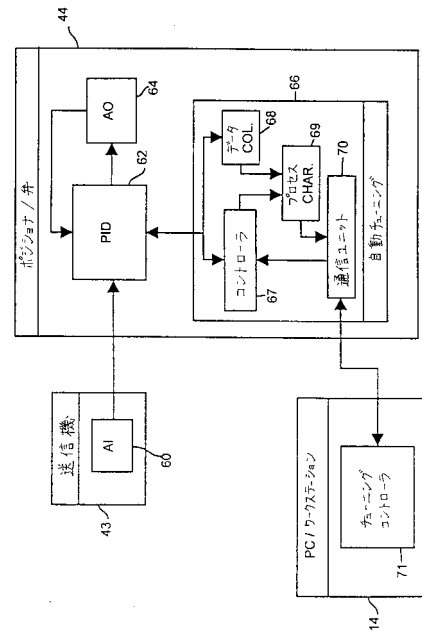
30

40

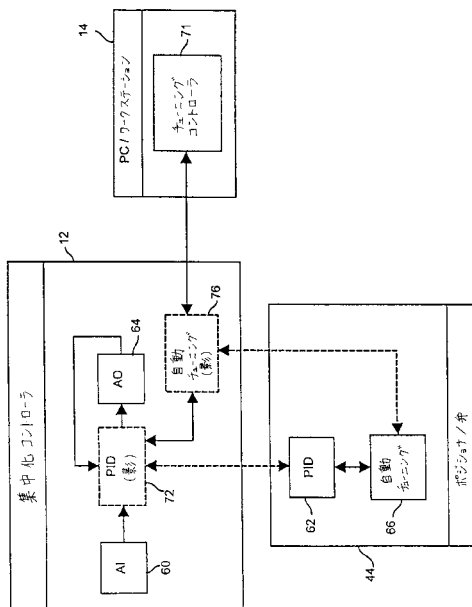
【図 1】



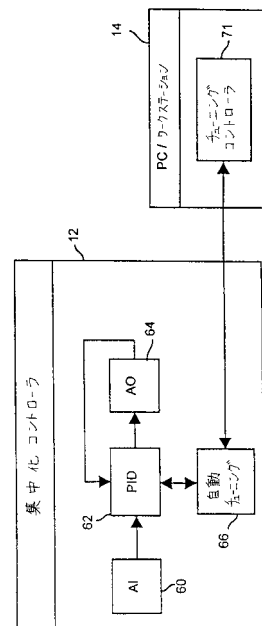
【図 2】



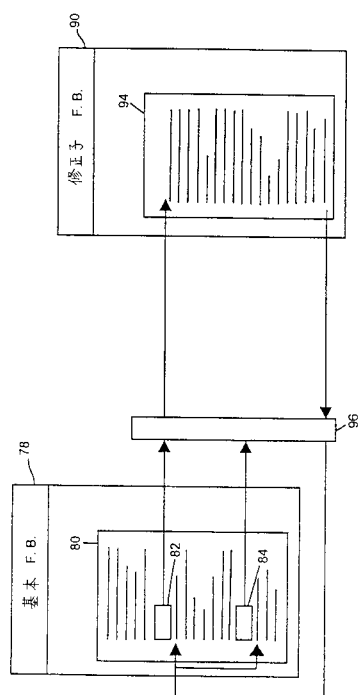
【図 3】



【図 4】



【図 5】



フロントページの続き

(72)発明者 デニス エル. スティーブソン
アメリカ合衆国 78681 テキサス ラウンド ロック セイバートゥース ドライブ 16
904

(72)発明者 ウィルヘルム ケー. ウォズニス
アメリカ合衆国 78681 テキサス ラウンド ロック ヒルサイド ドライブ 17004

合議体

審判長 千葉 成就

審判官 菅澤 洋二

審判官 刈間 宏信

(56)参考文献 国際公開第90/1183 (WO, A1)
特開平5-73104 (JP, A)
特開平10-91211 (JP, A)

(58)調査した分野(Int.Cl., DB名)
G05B13/02