

(19)日本国特許庁(JP)

## (12)特許公報(B2)

(11)特許番号

特許第7206581号

(P7206581)

(45)発行日 令和5年1月18日(2023.1.18)

(24)登録日 令和5年1月10日(2023.1.10)

(51)国際特許分類

G 0 5 B 11/42 (2006.01)

F I

G 0 5 B 11/42

Z

請求項の数 6 外国語出願 (全38頁)

(21)出願番号	特願2016-199737(P2016-199737)	(73)特許権者	512132022
(22)出願日	平成28年10月11日(2016.10.11)		フィッシャー・ローズマウント システ
(65)公開番号	特開2017-76392(P2017-76392A)		ムズ, インコーポレイテッド
(43)公開日	平成29年4月20日(2017.4.20)		アメリカ合衆国 テキサス 7 8 6 8 1 -
審査請求日	令和1年9月27日(2019.9.27)		7 4 3 0 ラウンド ロック ウェスト ル
審査番号	不服2021-16009(P2021-16009/J		イス ヘナ ブルバード 1 1 0 0 ビルデ
	1)		イング 1 エマーソン プロセス マネー
審判請求日	令和3年11月22日(2021.11.22)		ジメント
(31)優先権主張番号	62/240,322	(74)代理人	100079049
(32)優先日	平成27年10月12日(2015.10.12)		弁理士 中島 淳
(33)優先権主張国・地域又は機関	米国(US)	(72)発明者	テレース エル., プレビンズ
(31)優先権主張番号	15/081,189		アメリカ合衆国 テキサス州 7 8 6 8 1
(32)優先日	平成28年3月25日(2016.3.25)		ラウンド ロック キャメル ドライブ 3
(33)優先権主張国・地域又は機関			8 0 1
	最終頁に続く	(72)発明者	ベテ、 ストルテノウ
			最終頁に続く

(54)【発明の名称】 非周期的に更新されるコントローラにおける速度に基づく制御、プロセスを制御する方法、プロセスコントローラ

## (57)【特許請求の範囲】

## 【請求項1】

プロセスを制御する方法であって、

コンピュータ処理デバイス上で、制御信号を発生させるために、制御ルーチンの複数の繰り返しを実施することであって、前記制御ルーチンの各繰り返し中に、

コンピュータ処理デバイスを使用して、前記制御信号を生成する際に使用するための積分フィードバック寄与を発生させることであって、前記制御ルーチンの現在の繰り返しを含む前記複数の繰り返しの各々の間に、前記制御ルーチンの先行する繰り返しの積分フィードバック寄与値及び新しいプロセス変数測定値を受信した最後の繰り返し以来の全ての以前の繰り返しに対する前記制御信号の合計した値から、前記制御ルーチンの前記現在の繰り返しに対する現在の積分フィードバック寄与値を決定するために、連続的に更新される繰り返しフィルタを使用することを含む、発生させることと、

新しいプロセス変数測定値を受信する各コントローラ繰り返し中に、前記連続的に更新される繰り返しフィルタに接続されたスイッチによる判定に基づいて、前記制御ルーチンの前記現在の繰り返しに対する前記制御信号を発生させるために、前記現在の積分フィードバック寄与を使用し、かつ前記制御信号に対する新しいプロセス変数測定値を受信しないコントローラ繰り返し中に、前記スイッチによる判定に基づいて、前記制御信号を発生させるために、いかなる積分フィードバック寄与も使用しないか、ゼロ値を使用することと、を含む、実施することと、

前記プロセスを制御するために、前記制御信号を使用することと、を含む、方法。

10

20

**【請求項 2】**

前記制御信号を発生させるために、各繰り返し中に比例寄与を発生させ、各繰り返し中に前記比例寄与を使用することを更に含む、請求項 1 に記載の方法。

**【請求項 3】**

前記プロセス変数測定値が、前記制御信号の影響を受けているプロセスパラメータの測定値である、請求項 1 又は請求項 2 に記載の方法。

**【請求項 4】**

前記プロセスパラメータが、前記制御信号に応答するフィールドデバイスによって制御されるプロセス変数である、請求項 3 に記載の方法。

**【請求項 5】**

前記積分フィードバック寄与を決定することが、前記制御ルーチンの前記現在の繰り返しに対する前記制御信号と前記制御ルーチンの前記先行する繰り返しの前記積分フィードバック寄与値との間の差分に、リセット時間及びコントローラ実行期間に依存する係数を乗じたものに基づいて、前記積分フィードバック寄与値を発生させることを含む、請求項 1 ~ 請求項 4 の何れか 1 項に記載の方法。

**【請求項 6】**

制御信号を発生させるために制御ルーチンの複数の繰り返しの実施することが、

( i ) 前記制御ルーチンの各繰り返し中に、設定点値、直前のプロセス変数の測定値、及び前記積分フィードバック寄与に基づいて、前記制御信号を発生させること、及び

( i i ) 前記制御ルーチンの各繰り返し中に、設定点値、直前に受信したプロセス変数測定値、及び比例ゲイン値から比例成分を発生させることと、前記現在の繰り返し中に発生させた前記比例成分と、以前の繰り返し中に発生させた前記比例成分との間の差分を使用して、前記制御信号を発生させることと、

の少なくとも 1 つを含む、請求項 1 ~ 請求項 5 の何れか 1 項に記載の方法。

**【発明の詳細な説明】****【技術分野】****【0001】**

本特許は、非周期的制御または低速のフィードバックプロセス変数通信を使用する、プロセス制御システムにおける速度に基づく制御を提供するための方法及びシステムの補償に関し、より具体的には、プロセス動力学と比較して低速なレートでプロセス変数フィードバックを受信するときに、制御を実施するときのプロセスをロバストに制御するように構成されるデバイス及び方法に関する。

**【背景技術】****【0002】**

化学、石油、または他のプロセスで使用されるような分散型または拡張可能なプロセス制御システムなどのプロセス制御システムは、典型的に、アナログバス、デジタルバス、またはアナログ及びデジタルを組み合わせたバスを介して、相互に、少なくとも 1 つのホストまたはオペレータワークステーションに、及び 1 つ以上のフィールドデバイスに通信可能に連結される、1 つ以上のプロセスコントローラを含む。例えば、弁、弁ポジション、スイッチ、及び送信機（例えば、温度、圧力、及び流量センサ）であり得るフィールドデバイスは、弁を開くまたは閉じること、及びプロセスパラメータを測定することなどのプロセス内の機能を行う。プロセスコントローラは、フィールドデバイスによって作成されるプロセス測定値及び/またはフィールドデバイスに関する他の情報を示す信号を受信し、この情報を使用して、制御ルーチンを実施して制御信号を発生させ、該制御信号は、プロセスの動作を制御するために、バスを通じてフィールドデバイスに送信される。フィールドデバイス及びコントローラからの情報は、典型的に、オペレータワークステーションによって実行される 1 つ以上のアプリケーションが利用することができる。オペレータが、プロセスの現在の状態を確認すること、プロセスの動作を修正することなどの、プロセスに関する任意の所望の機能を行うことを可能にする。

**【0003】**

Emerson Process Managementによって販売されるDelta V（登録商標）システムなどの、いくつかのプロセス制御システムは、コントローラまたは異なるフィールドデバイスに位置する、モジュールと称される機能ブロックまたは一群の機能ブロックを使用して、制御動作及び／または監視動作を行う。これらの事例において、コントローラまたは他のデバイスは、1つ以上の機能ブロックまたはモジュールを含み、かつ実行することができ、そのそれぞれが、他の機能ブロック（同じデバイス内または異なるデバイス内のいずれか）からの入力を受信し、及び／または他の機能ブロックに出力を提供し、プロセスパラメータの測定若しくは検出、デバイスの監視、デバイスの制御、または比例 - 積分 - 微分（PID）制御ルーチンの実施などの制御動作の実行などの、何らかのプロセス動作を行う。プロセス制御システム内の異なる機能ブロック及びモジュールは、全般的に、（例えば、バスを通じて）互いに通信して、1つ以上のプロセス制御ループを形成するように構成される。

10

#### 【0004】

プロセスコントローラは、典型的に、プロセスに関して定義されるか、または、フロー制御ループ、温度制御ループ、圧力制御ループなどのプロセス内に含まれる、多数の異なるループの各々に対して、異なるアルゴリズム、サブルーチン、または制御ループ（これらは全て制御ルーチンである）を実行するようにプログラムされる。全般的に言えば、そのような各制御ループは、アナログ入力（AI）機能ブロックなどの1つ以上の入力ブロック、比例 - 積分 - 微分（PID）またはファジー論理制御ブロックなどの単一出力の制御ブロック、及びアナログ出力（AO）機能ブロックなどの出力ブロックを含む。制御ルーチン、及びそのようなルーチンを実施する機能ブロックは、PID制御、ファジー論理制御、及びスミス予測器またはモデル予測制御（MPC）などのモデルベースの技法を含む多数の制御技法、に従って構成されている。

20

#### 【0005】

ルーチンの実行をサポートするために、典型的な工業プラントまたはプロセスプラントは、1つ以上のプロセスコントローラ及びプロセスI/Oサブシステムと通信可能に接続される集中制御室を有し、プロセスコントローラ及びプロセスI/Oサブシステムは、1つ以上のフィールドデバイスに接続される。伝統的に、アナログフィールドデバイスは、信号伝送及び電力供給の双方のための二線式または四線式の電流ループによってコントローラに接続されている。制御室に信号を送信するアナログフィールドデバイス（例えば、センサまたは電送機）は、電流が検知プロセス変数に比例するように、電流ループを流れる電流を変調する。一方で、制御室の制御下でアクションを行うアナログフィールドデバイスは、ループを通る電流の大きさによって制御される。

30

#### 【0006】

データ転送量の増加に伴って、プロセス制御システム設計の1つの特に重要な態様は、フィールドデバイスが、プロセス制御システムまたはプロセスプラント内のコントローラに、及び他のシステムまたはデバイスに互いに通信可能に連結される様式を含む。全般に、フィールドデバイスがプロセス制御システム内で機能することを可能にする様々な通信チャンネル、リンク、及び経路は、一般的に、集成的に入力／出力（I/O）通信ネットワークと称される。

40

#### 【0007】

I/O通信ネットワークを実施するために使用される通信ネットワークトポロジー及び物理的な接続または経路は、特にネットワークが有害な環境要因または厳しい条件を受けるときに、フィールドデバイス通信のロバスト性または健全性にかなりの影響を及ぼすことがあり得る。これらの要因及び条件は、1つ以上のフィールドデバイス、コントローラなどの間の通信の健全性を損ない得る。コントローラとフィールドデバイスとの間の通信は、監視アプリケーションまたは制御ルーチンが、典型的に、ルーチンの各繰り返しに対するプロセス変数の周期的な更新を必要とするので、任意のそのような途絶に特に敏感である。したがって、損なわれた制御通信は、プロセス制御システムの効率性及び／または利益性の低下、及び過剰な摩耗または装置に対する損傷、並びに任意の数の潜在的に有害

50

な故障をもたらし得る。

【 0 0 0 8 】

ロバストな通信を保証するために、プロセス制御システムで使用する I / O 通信ネットワークは、歴史的にハードワイヤードである。残念なことに、ハードワイヤードネットワークは、数多くの複雑性、課題、及び制限を導入する。例えば、ハードワイヤードネットワークの品質は、時間とともに劣化し得る。更に、特に、I / O 通信ネットワークが、例えば数エーカーの土地を費やす精油所または化学プラントといった広い面積にわたって分散される大規模な工業プラントまたは施設と関連付けられる場合に、ハードワイヤード I / O 通信ネットワークは、典型的に、設置に費用がかかる。必要な長い配線取り付けは、典型的に、かなりの労働力、材料、及び費用を伴い、また、配線インピーダンス及び電磁妨害から生じる信号劣化を導入する場合がある。これらの及び他の理由により、ハードワイヤード I / O 通信ネットワークは、全般的に、再構成、修正、または更新することが困難である。

10

【 0 0 0 9 】

無線 I / O 通信ネットワークを使用して、ハードワイヤード I / O 通信ネットワークと関連付けられる問題点のいくつかを軽減することが提案されてきた。例えば、その開示全体が参照により本明細書に明示的に組み込まれる、米国特許出願公開第 2 0 0 3 / 0 0 4 3 0 5 2 号、名称「Apparatus for Providing Redundant Wireless Access to Field Devices in a Distributed Control System」は、ハードワイヤード通信の使用を増やす、または補うために無線通信を利用するシステムを開示している。

20

【 0 0 1 0 】

制御関連の伝送のための無線通信に対する依存は、数ある中でも信頼性が懸念されるため、伝統的に制限される。上で説明されるように、最新の監視アプリケーション及びプロセス制御は、最適な制御レベルを達成するために、コントローラとフィールドデバイスとの間の信頼性のあるデータ通信に依存する。更に、典型的なコントローラは、そのプロセスにおいて望ましくない偏差を迅速に修正するために、高速で制御アルゴリズムを実行する。望ましくない環境要因または他の悪条件は、監視及び制御アルゴリズムのそのような実行をサポートするために必要な高速通信を妨害または阻止する間欠的干渉を生じさせる場合がある。幸運にも、無線ネットワークは、過去 2 0 年 にわたってはるかにロバストになり、いくつかのタイプのプロセス制御システムにおいて無線通信の信頼性のある使用を可能にしてきた。

30

【 0 0 1 1 】

しかし、電力消費は、依然として、プロセス制御アプリケーションで無線通信を使用するときの、複雑化要因である。無線フィールドデバイスが I / O ネットワークから物理的に分離されるので、フィールドデバイスは、典型的に、それ自体の電力源を提供する必要がある。故に、フィールドデバイスは、バッテリー式とするか、太陽エネルギーを引き出すか、または振動、熱、圧力などの環境エネルギーを取り出すことができる。これらのデバイスにとって、データ伝送のために消費されるエネルギーは、総エネルギー消費のかなりの部分を占める場合がある。実際に、測定されているプロセス変数を検知または検出するためになされるステップなどのフィールドデバイスによって行われる他の重要な動作中よりも、無線通信接続を確立し、維持するプロセス中のほうが、より多くの電力を消費する場合がある。無線プロセス制御システムの電力消費を低減させ、したがって、バッテリー寿命を延ばすために、センサなどのフィールドデバイスが非周期的な、低速の、または間欠的な様式でコントローラと通信する無線プロセス制御システムを実施することが提案されてきた。1つの事例において、フィールドデバイスは、プロセス変数のかなりの変化が検出され、コントローラと非周期的通信につながるときにだけ、コントローラと通信、またはコントローラに対してプロセス変数の測定値を送信することができる。

40

【 0 0 1 2 】

非周期的プロセス変数測定値更新を取り扱うために開発された 1 つの制御技法は、低頻

50

度なまたは非周期的な測定値更新の間にコントローラによって生成される制御信号に対して予想されるプロセス応答の指示を提供し、維持する制御システムを使用する。予想されるプロセス応答は、所与の測定値更新のための制御信号に対して予想されるプロセス応答を算出する数学的モデルによって開発することができる。この技法の一例は、その開示全体が参照により本明細書に明示的に組み込まれる、米国特許第7,587,252号、名称「Non-Periodic Control Communications in Wireless and Other Process Control Systems」で説明されている。具体的には、この特許は、非周期的プロセス変数測定値更新を受信すると、制御信号に対して予想されるプロセス応答の指示を生成し、次の非周期的プロセス変数測定値更新の到着まで、予想されるプロセス応答の生成された指示を維持するフィルタを有する、制御システムを開示している。別の例として、その開示全体が参照により本明細書に明示的に組み込まれる、米国特許第7,620,460号、名称「Process Control With Unreliable Communication」は、制御信号に対して予想される応答の指示を提供するが、最後の非周期的測定値更新以来経過した時間の測定値を組み込んで、予想されるプロセス応答のより正確な指示を生成するために、フィルタを更に修正するフィルタを含む、システムを開示する。

#### 【0013】

しかしながら、多くの制御アプリケーションにおいて、プロセス制御システムは、プロセス動作中に、設定点変化を受信する場合がある。全般的に、周期的に更新される制御システム（例えば、ハードワイヤード制御通信システム）の実行中に、設定点が変化したときに、設定点と測定されたプロセス変数との間のエラーに対する比例アクションを行うように設計されるコントローラは、新しい定常状態値に向かってプロセス変数を駆動するために、コントローラ出力を直ちに变化させる。しかしながら、上述の双方の例において説明されるように動作する、低頻度で非周期的な測定値更新を受信する無線制御システムにおいて、各新しい測定値更新によって反映される、測定されたプロセス応答は、最後の測定値更新を受信した後しばらくして行われる設定点変化によって生じる出力の変化に加えて、最後の測定値更新のために取り込まれるコントローラ出力において行われる変化を反映する。この事例において、（米国特許第7,620,460号で説明されるように）コントローラ出力、及び最後の測定値更新以来の時間に基づいてコントローラのリセット成分を算出することは、最後の測定値更新の後に行われる変化を過剰に補償する場合がある。したがって、設定点変化に対するプロセス応答は、最後の測定値更新の後に設定点変化がいつ行われたかに基づいて、異なる場合がある。結果として、このシステムは、コントローラが設定点の変化後に制御信号を発生させるときに、以前に発生させた（及び現在古くなった）予想される応答の指示に依存し続けるので、設定点変化に対して迅速に、またはロバストに応答しない。この問題を解決するために、その開示全体が参照により明示的に組み込まれる、米国特許出願公開第2013/0184837号、名称「Compensating for Setpoint Changes in a Non-Periodically Updated Controller」は、いかなる新しいプロセス変数測定値も受信しない時間中に、制御された変数の動作を追跡するために、コントローラ内のフィードバックループにおいて連続的に更新されるフィルタを使用し、また、コントローラで新しいプロセス変数測定値を受信したときに、このコントローラの出力を使用するが、別様には、プロセス変数の測定値を受信して制御信号を発生させた最も近時からのフィルタの出力を使用する、システムを開示している。このシステムにおいて、発生させた制御信号は、プロセス変数の測定値を受信する時間の間にプロセス変数の設定点が変化したときに、より良好に応答し、より良好に動作する。

#### 【0014】

更に、バッテリー駆動の送信機を無線制御システムにおいて使用するときには、長時間のバッテリー寿命を維持するようにシステムをセットアップすることが望ましい。例えば、現在の送信機及びバッテリー技術を使用して3～5年のバッテリー寿命を得るためには、全般的に、8秒以上の通信更新レートを使用することが必要である。しかしながら、プロセスの

10

20

30

40

50

制御を維持するために、プロセスの応答時間と関連付けられるレートの少なくとも4倍の時間（すなわち、プロセス応答時間の逆数）であるレートで受信するプロセスフィードバック測定値をそれでも有することが重要であるので、そのような低速の更新レートを使用することは、30秒以上のプロセス応答時間を有するプロセスに対するPID（比例・積分・微分）に基づく無線制御の使用を制限する。

#### 【0015】

なお更に、移動すべき最終的な位置を示す、弁または他の被制御要素に提供される位置出力、例えば4~20mAの信号またはデジタル信号を生成することを使用するものを含む、無線制御に対処するために使用することができる、様々なタイプのPIDアルゴリズムが存在する。しかしながら、例えばある特定の時間量にわたって移動可能な要素に通電することによって、弁または他の被制御要素をある特定の方向にある特定の量だけ移動させるように指示する、速度に基づく制御信号を提供する、他のPIDアルゴリズムも存在する。そのような速度に基づく制御信号は、一般に、電気モーターとともに使用され、また、（特定の期間にわたって移動させるために移動可能な要素に通電しなければならない、弁の時間量を示すように変調されるパルス幅を有する）パルス信号の形態で制御信号を提供する。速度に基づくコントローラは、移動可能な要素によって取得される実際の位置を示す信号ではなく、位置信号の変化を生成する傾向がある。したがって、速度に基づく制御アルゴリズムは、増分（増加/減少）出力をアクチュエータに提供するために使用される傾向があり、よって、位置フィードバックを提供することができないアクチュエータを制御する際に使用することができる。

#### 【発明の概要】

#### 【0016】

制御技法は、制御されたプロセス変数の測定値がプロセスコントローラへのフィードバックとして提供されるレートに関して高速な動力学を有する、プロセスまたは制御ループのロバストな制御を可能にする。具体的には、本制御技法は、位置または速度の形態のPIDアルゴリズムを使用して、プロセスを制御することができ、プロセス変数の測定値またはフィードバック信号が、プロセス応答時間に等しいか、更にはそれよりも長い時間間隔でコントローラに提供される。具体的には、本制御技法は、フィードバック時間間隔よりも2~4倍短い応答時間を有するプロセスにおいて、ロバストな制御を提供するために使用することができる。そのような状況は、例えば、プロセス変数のフィードバック測定値が、無線で、間欠的に、またはプロセスの応答時間よりも短いか、応答時間に近い、若しくは応答時間よりも長い時間間隔でコントローラに提供される無線制御を使用するときにより起こり得る。

#### 【0017】

開示される速度PID制御ルーチンは、アクチュエータが増分入力を必要とし、かついかなる位置フィードバックもコントローラに利用できないときに無線測定値を使用して制御を行うために、位置または増分入力を必要とするアクチュエータにインターフェースするときに有線測定値によって制御を行うために、及び従来の設置物並びに無線測定値が制御に使用される新しい設置物に対処するために、などの、数多くの異なる状況において使用することができる。更に、速度の形態のPIDアルゴリズムの調整は、プロセスゲイン及び動力学に基づいて行うことができ、また、無線通信レートから独立している。なお更に、速度の形態のPID制御ルーチンは、通信の喪失時に最後の出力位置を自動的に保持し、通信が再確立されたときにバンプレスな復旧を提供する。

#### 【0018】

1つの事例において、新しい制御技法を実施するコントローラは、全般的に、異なる構造を含み該構造では、差分、比例、積分、及び微分制御信号成分を発生させ、それらを使用して、差分または移動に基づく制御信号を生成し、該信号は、次いで、被制御デバイス、例えば弁に送信されて、被制御デバイスの動作、したがってプロセスを制御する。この差分または速度に基づく制御形態は、低速なプロセス変数のフィードバック測定値の存在下で、標準的なPID制御よりも良好に動作する制御信号を発生させる。具体的には、こ

10

20

30

40

50

の制御技法を使用するコントローラは、各コントローラの繰り返し中に、以前の比例制御信号成分と新しく算出された比例制御信号成分との間の差分を表す、差分の比例値を発生させ、この差分の比例値は、コントローラからの各新しい制御信号の基準として使用される。しかしながら、コントローラで新しいプロセス変数測定値信号が利用することができるときに、様々な他の制御信号成分、例えば微分制御信号成分及び/または積分制御信号成分を、差分の比例制御信号成分に加えること、または該成分と組み合わせることができる。これらの2つの制御信号成分はまた、新しく算出された値と以前に算出された値との間の差分に基づくこともできる。具体的には、新しい微分成分は、コントローラ繰り返し中に算出することができ、その時点で、プロセス変数測定値の新しく受信した値をコントローラで利用することができる。同様に、新しい積分成分は、連続的に更新されるフィルタを使用して発生させることができ、該フィルタは、コントローラの各制御ルーチンの繰り返しに対するプロセスの予想される応答の新しい指示を発生させる。しかしながら、連続的に更新されるフィルタの出力は、プロセス変数測定値の新しい値を受信したときにだけ、新しい積分要素を発生させるために使用される。それ以外のときには、積分制御信号成分は、ゼロに設定される。

10

【0019】

本明細書で開示される速度に基づくPID制御技法は、差分の形態を使用して、（プロセス変数のフィードバック信号がコントローラの入力に存在する時間の間であっても）設定点変化に迅速に適合する制御信号を生成するが、それでも、制御されているプロセスの応答時間の逆数よりも遅い（例えば、それよりも2～4倍も遅い）、応答時間の逆数に近い、または応答時間の逆数よりも速いレートで受信されるフィードバック信号を含む、低速に受信される（例えば、間欠的な）フィードバック信号の存在下で、ロバストで安定した制御を提供する。

20

【図面の簡単な説明】

【0020】

【図1】例示的な、周期的に更新されるハードワイヤードプロセス制御システムのブロック線図である。

【図2】プロセス応答時間を含む、例示的な、周期的に更新されるハードワイヤードプロセス制御システムのプロセス入力に対するプロセス出力応答を例示するグラフである。

【図3】低速または非周期的フィードバック入力を受信するコントローラを有する無線プロセス制御システムの実施例を例示するブロック線図である。

30

【図4A】周期的に更新されるハードワイヤードプロセス制御システムにおける設定点変化またはフィードフォワード外乱のロバストな補償を可能にする、例示的なコントローラのブロック線図である。

【図4B】コントローラがいくつかの設定点変化に反応するときの、図4Aの例示的なコントローラのプロセス出力応答を例示するグラフである。

【図5】コントローラがフィードバック信号におけるプロセス及び/または測定値の遅延を補償する、非周期的に更新されるプロセス制御システムにおいて設定点変化の補償を行う例示的なコントローラのブロック図である。

【図6】プロセスコントローラが微分寄与またはレート寄与を使用して制御信号を決定する、非周期的に更新されるプロセス制御システムにおいて設定点変化の補償を行う例示的なコントローラのブロック図である。

40

【図7】プロセスコントローラが、フィールドデバイス、制御要素、または他の下流のデバイスから提供される追加的なコントローラ入力データを受信して、プロセスの動作の応答に影響を及ぼす、非周期的に更新されるプロセス制御システムにおいて設定点変化の補償を行う例示的なコントローラのブロック図である。

【図8】プロセスコントローラが、フィールドデバイスに対する実際または暗黙いずれかのコントローラ入力データの使用に適応する、非周期的に更新されるプロセス制御システムにおいて設定点変化の補償を行う例示的なコントローラのブロック図である。

【図9】低速に受信したプロセス変数測定値信号に反応して、プロセス制御システムにお

50

ける設定点変化またはフィードフォワード外乱のロバストな補償を可能にする、例示的な速度に基づくPIDコントローラのブロック図である。

【図10】低速に受信したプロセス変数測定値信号にตอบสนองして、プロセス制御システムにおける設定点変化またはフィードフォワード外乱のロバストな補償を可能にする、例示的な速度に基づくPIDコントローラのブロック図である。

【図11A】一次制御変数における設定点変化にตอบสนองし、8秒のプロセス応答時間を有する、有線及び無線双方の構成における、例示的な従来技術の速度に基づくPIDコントローラのシミュレーションされたプロセス応答を例示するグラフである。

【図11B】一次制御変数における設定点変化にตอบสนองし、8秒のプロセス応答時間を有する、有線及び無線双方の構成における、例示的な本発明による速度に基づくPIDコントローラのシミュレーションされたプロセス応答を例示するグラフである。

10

【図12A】一次制御変数における設定点変化にตอบสนองし、3秒のプロセス応答時間を有する、有線及び無線双方の構成における、例示的な従来技術の速度に基づくPIDコントローラのシミュレーションされたプロセス応答を例示するグラフである。

【図12B】一次制御変数における設定点変化にตอบสนองし、3秒のプロセス応答時間を有する、有線及び無線双方の構成における、例示的な本発明による速度に基づくPIDコントローラのシミュレーションされたプロセス応答を例示するグラフである。

【図13A】外乱変化にตอบสนองし、8秒のプロセス応答時間を有する、有線及び無線双方の構成における、例示的な従来技術の速度に基づくPIDコントローラのシミュレーションされたプロセス応答を例示するグラフである。

20

【図13B】外乱変化にตอบสนองし、8秒のプロセス応答時間を有する、有線及び無線双方の構成における、例示的な本発明による速度に基づくPIDコントローラのシミュレーションされたプロセス応答を例示するグラフである。

【図14A】外乱変化にตอบสนองし、3秒のプロセス応答時間を有する、有線及び無線双方の構成における、例示的な従来技術の速度に基づくPIDコントローラのシミュレーションされたプロセス応答を例示するグラフである。

【図14B】外乱変化にตอบสนองし、3秒のプロセス応答時間を有する、有線及び無線双方の構成における、例示的な本発明による速度に基づくPIDコントローラのシミュレーションされたプロセス応答を例示するグラフである。

【発明を実施するための形態】

30

【0021】

制御技法は、プロセス測定値フィードバック信号が低速または間欠的に受信されるプロセスループにおいて制御を行うために使用することができ、特に、プロセス測定値フィードバック信号を、プロセス応答時間の逆数などの、制御されているプロセス動力学と関連付けられるレートよりも遅い、レートと同程度の、レートよりも僅かに速いレートで受信するときに有用である。このコントローラは、例えば、低速な及び/または非周期的様式で、特に、制御されているプロセス動力学のプロセス応答レート（すなわち、制御されているプロセス変数に対するプロセス応答時間の逆数）未満である、プロセス応答レートと同程度である、または別様にはプロセス応答レートに類似するレートで、プロセス測定値信号をフィードバック信号として受信するコントローラにおいて使用することができる。

40

1つの事例において、制御技法は、比例寄与信号を微分寄与信号及び積分寄与信号のうちの1つ以上と組み合わせることによって、弁などのプロセスデバイスを制御する際に使用するための制御信号を生成する。速度に基づくPIDアルゴリズムが使用される1つの事例において、コントローラは、設定点と、直前に受信したプロセス変数測定値のフィードバック信号との間の差分から、比例寄与値を生成する。この誤差信号は、次いで、ゲイン信号を乗じて、コントローラの最後の実行サイクル以来のこの信号の変化を決定する差分ユニットに提供される。更に、微分ユニットは、誤差信号を受信し、基本的にはコントローラで最後の測定信号を受信して以来の誤差信号を経時的に微分して、誤差信号に対する微分算出を行うことができる。微分モジュールまたは算出の出力はまた、変化検出ユニットまたは差分ユニットにも提供され、微分算出の現在の出力と、制御信号の算出に使用さ

50



れる以前の値との間の差分を決定する。同様に、積分算出ユニットが、制御信号の差分を比例ユニット及び微分ユニットによる生成物として受信し、積分する（例えば、合計する）。加算器の出力は、加算器の出力をフィルタ処理して積分寄与信号を生成するために、フィルタに提供される。しかしながら、この積分寄与は、加算器に提供され、新しいフィードバック値がコントローラに提供されたときにだけ、この寄与と制御信号の出力とを合計する。すなわち、積分寄与は、全てのコントローラ繰り返しに対してゼロに設定されるが、コントローラで新しいフィードバック信号を利用することができるものは除く。

#### 【 0 0 2 2 】

全般的に言えば、コントローラ内の積分寄与ユニットの連続的に更新されるフィルタは、フィールドデバイスからプロセス変数測定値更新を低速または非周期的に受信するにもかかわらず、コントローラの各制御ルーチン繰り返し中に、予想されるプロセス応答（フィードバック寄与とも呼ばれる）の指示を発生させる。連続的に更新されるフィルタは、部分的には、各制御ルーチン繰り返し中に、予想される応答の指示を発生させるために、最後の制御ルーチン繰り返し及び制御ルーチン実行期間からの以前に発生させた予想される応答の指示を使用する。加えて、コントローラ内の積分出力スイッチは、連続的に更新されるフィルタの出力を、最新の測定値の指示に基づく積分（リセットとしても知られる）寄与などのフィードバック寄与として、制御信号に提供する。全般的に言えば、積分出力スイッチは、新しい測定値信号を利用することができないコントローラの各繰り返し中の、制御信号に対する積分またはリセット寄与として、速度の形態のPIDコントローラにおいて、最後の測定値更新をコントローラによって受信した時間に連続的に更新されるフィルタによって発生させられるような予想されるプロセス応答、またはゼロ値を提供する。新しい測定値更新を利用することができるときに、積分出力スイッチは、（新しい測定値更新の指示に基づいて）連続的に更新されるフィルタによって発生させた、予想されるプロセス応答の新しい指示にクランプし、制御信号の積分寄与として、新しい予想されるプロセス応答を提供する。その結果、コントローラは、連続的に更新されるフィルタを使用して、各コントローラ繰り返し中に、プロセスの新しい予想される応答を決定し、各新しい予想されるプロセス応答は、コントローラで新しい測定値を利用することができるときにだけ制御信号の積分またはリセット成分が変化する場合であっても、測定値更新の間の時間に行われたものであり、また、制御信号の展開中にコントローラ出力に影響を及ぼす、設定点変化またはフィードフォワード変化の影響を反映する。

#### 【 0 0 2 3 】

図1には、説明される制御方法を実施するために使用することができる、プロセス制御システム10が例示される。プロセス制御システム10は、データヒストリアン12に、及び1つ以上のホストワークステーションまたはコンピュータ13（任意のタイプのパーソナルコンピュータ、ワークステーションなどとして行うことができる）に接続することができる。各々が、表示画面14を有する、プロセスコントローラ11を含む。コントローラ11はまた、入力/出力（I/O）カード26及び28を介して、フィールドデバイス15～22にも接続される。データヒストリアン12は、データを記憶するための任意の所望のタイプのメモリ、及び任意の所望のまたは既知のソフトウェア、ハードウェア、またはファームウェアを有する、任意の所望のタイプのデータ収集ユニットとすることができる。図1において、コントローラ11は、ハードワイヤード通信ネットワーク及び通信スキームを使用して、フィールドデバイス15～22に通信可能に接続される。

#### 【 0 0 2 4 】

全般的に、フィールドデバイス15～22は、センサ、弁、送信機、位置決め器などの任意のタイプのデバイスとすることができる。一方で、I/Oカード26及び28は、任意の所望の通信またはコントローラプロトコルに準拠する任意のタイプのI/Oデバイスとすることができる。コントローラ11は、メモリ24に記憶される1つ以上のプロセス制御ルーチン（または任意のモジュール、ブロック、またはそのサブルーチン）を実施または監督する、プロセッサ23を含む。全般的に言えば、コントローラ11は、任意の所望の様式でプロセスを制御するために、デバイス15～22、ホストコンピュータ13、及

10

20

30

40

50

びデータヒストリアン 12 と通信する。更に、コントローラ 11 は、一般に機能ブロックと称されるものを使用して、制御ストラテジまたはスキームを実施し、各機能ブロックは、プロセス制御システム 10 内でプロセス制御ループを実施するために、（リンクと呼ばれる通信を介して）他の機能ブロックとともに動作する、制御ルーチン全体のオブジェクトまたは他の部分（例えば、サブルーチン）である。機能ブロックは、典型的に、プロセス制御システム 10 内でいくつかの物理的機能を行うために、送信機、センサ、若しくは他のプロセスパラメータ測定デバイスと関連付けられるような入力機能、PID、ファジー論理制御などを行う制御ルーチンと関連付けられるような制御機能、または弁などのいくつかのデバイスの動作を制御する出力機能、のうちの 1 つを行う。当然、ハイブリッド及び他のタイプの機能ブロックが存在し、本明細書で利用することができる。機能ブロックは、下で説明されるように、コントローラ 11 または他のデバイスに記憶し、それによって実行することができる。

#### 【0025】

図 1 の分解ブロック 30 で例示されるように、コントローラ 11 は、制御ルーチン 32 及び 34 として例示される、いくつかの単一ループ制御ルーチンを含むことができ、また、所望であれば、制御ループ 36 として例示される、1 つ以上のアドバンスト制御ループを実施することができる。各制御ループは、典型的に、制御モジュールと称される。単一ループ制御ルーチン 32 及び 34 は、弁などのプロセス制御デバイス、温度及び圧力送信機などの測定デバイス、またはプロセス制御システム 10 内の任意の他のデバイスと関連付けることができる、適切なアナログ入力（AI）及びアナログ出力（AO）機能ブロックにそれぞれ接続される単一入力／単一出力ファジー論理制御ブロック及び単一入力／単一出力 PID 制御ブロックを使用して、単一ループ制御を行うように例示される。アドバンスト制御ループ 36 は、1 つ以上の AI 機能ブロックに通信可能に接続される入力及び 1 つ以上の AO 機能ブロックに通信可能に接続される出力を有する、アドバンスト制御ブロック 38 を含むように例示されるが、アドバンスト制御ブロック 38 の入力及び出力は、他のタイプの入力を受信し、他のタイプの制御出力を提供するために、任意の他の所望される機能ブロックまたは制御要素に接続することができる。アドバンスト制御ブロック 38 は、任意のタイプの多重入力、多重出力制御スキームを実施することができ、また、モデル予測制御（MPC）ブロック、ニューラルネットワークモデリングまたは制御ブロック、多変数ファジー論理制御ブロック、リアルタイムオブティマイザブロックなどを構成するか、または含むことができる。アドバンスト制御ブロック 38 を含む、図 1 に例示される機能ブロックは、スタンドアロンのコントローラ 11 によって実行することができ、または代替的に、ワークステーション 13 のうちの 1 つまたはフィールドデバイス 19 ~ 22 のうちの 1 つなどの、プロセス制御システムの任意の他の処理デバイスまたは制御要素の中に位置させて、実行することができることが理解されるであろう。一例として、フィールドデバイス 21 及び 22 は、それぞれ、送信機及び弁とすることができ、また、制御ルーチンを実施するための制御要素を実行することができ、よって、1 つ以上の機能ブロックなどの、制御ルーチンの一部を実施するための処理構成要素及び他の構成要素を含むことができる。より具体的には、図 1 に例示されるように、フィールドデバイス 21 は、アナログ入力ブロックと関連付けられる論理及びデータを記憶するためのメモリ 39 A を有することができ、一方で、フィールドデバイス 22 は、アナログ出力（AO）ブロックと通信している PID または他の制御ブロックと関連付けられる論理及びデータを記憶するためのメモリ 39 B を有するアクチュエータを含むことができる。

#### 【0026】

図 2 のグラフは、全般的に、制御ループ 32、34、及び 36（並びに / またはフィールドデバイス 21 及び 22 若しくは他のデバイスの中に存在する機能ブロックを組み込んでいる任意の制御ループ）のうちの 1 つ以上の実施に基づいて、プロセス制御システムのプロセス入力に応答して展開される、プロセス出力を例示する。実施されている制御ルーチンは、全般的に、図 2 において太い矢印 40 による時間軸に沿って示されている制御ルーチンの実行回数によって、多数のコントローラ繰り返しを通じて、周期的様式で実行す

10

20

30

40

50

る。従来の事例において、各制御ルーチン繰り返し40は、例えば送信機または他のフィールドデバイスによって提供される、細い矢印42によって示される更新されたプロセス測定値によってサポートされる。図2に例示されるように、典型的には、周期的制御ルーチン実行時間40の各々の間に制御ルーチンによってなされ、受信される、複数の周期的プロセス測定値42が存在する。測定値と制御の実行との同期化と関連付けられる制限を回避するために、多くの既知のプロセス制御システム（または制御ループ）は、プロセス可変測定値を2～10倍の係数でオーバーサンプリングするように設計される。そのようなオーバーサンプリングは、各制御ルーチンの実行または繰り返しの間に制御スキームにおいて使用するためのプロセス変数測定値が最新であることを確実にすることを補助する。また、制御の変動を最小にするために、従来の設計では、フィードバックに基づく制御が、プロセス応答時間よりも4～10倍速く実行されるべきであること、及び新しいプロセス変数測定値を各コントローラ繰り返し時に利用することができることが定められている。プロセス応答時間は、プロセス時定数（ ）（例えば、プロセス変数の変化の63%）に、プロセス入力ステップ変化44の実施後のプロセス遅延またはむだ時間（ $T_D$ ）（図2の下側の線45で示される）を加えたものと関連付けられる時間であるように、図2のグラフのプロセス出力応答曲線43に表される。いずれにしても、これらの従来の設計要件を満たすために、プロセス測定値更新（図2の矢印42によって示される）は、制御ルーチン実行レート（図2の矢印40によって示される）よりもはるかに速い、よって、プロセス応答時間よりもはるかに速いか、または高いレートでサンプリングされ、コントローラに提供される。

10

20

#### 【0027】

しかしながら、プロセスから高頻度で周期的な測定値のサンプルを取得することは、例えば、コントローラが1つ以上のフィールドデバイスから測定値を無線で受信するプロセス制御環境においてコントローラが動作しているときには、実用的でなく、または不可能な場合さえある。具体的には、これらの事例において、コントローラは、（無線センサ/送信機のバッテリー寿命を節約するために）低速なプロセス変数測定値または非周期的プロセス変数測定値を受信することだけしかできない場合がある。更に、これらの事例において、非周期的、更には周期的プロセス変数測定値の間の時間は、制御ルーチン実行レート（図2の矢印40によって示される）よりも長くなる場合がある。図3は、プロセス制御データの低速な及び/または非周期的無線通信、またはコントローラ11でのプロセス変数測定値の使用を実施することができる、例示的な無線プロセス制御システム10を描写する。

30

#### 【0028】

図3の制御システム10は、図1の制御システム10に本質的に類似し、同じ要素には同じ番号が付される。しかしながら、図3の制御システム10は、コントローラ11に、及び潜在的に互いに無線で通信可能に連結される、多数のフィールドデバイス60～64及び71を含む。図3に例示されるように、無線で接続されるフィールドデバイス60は、アンテナ65に接続され、アンテナ74と無線で通信するように協働し、次に該アンテナが無線I/Oデバイス68に連結される。更に、フィールドデバイス61～64は、有線-無線変換ユニット66に接続され、次に該ユニットがアンテナ67に次第に接続される。フィールドデバイス61～64は、アンテナ67を通して、更なる無線I/Oデバイス70に接続されるアンテナ73と無線で通信する。同じく図3に例示されるように、フィールドデバイス71は、アンテナ73及び74のうちの一方または双方と通信するアンテナ72を含み、それによって、I/Oデバイス68及び/または70と通信する。次にI/Oデバイス68及び70は、有線バックプレーン接続（図3に図示せず）を介して、コントローラ11に通信可能に接続される。この事例において、フィールドデバイス15～22は、I/Oデバイス26及び28を介して、コントローラ11にハードワイヤードされたままである。

40

#### 【0029】

図3のプロセス制御システム10は、下で説明されるように、全般的に、送信機60～

50

6 4 またはフィールドデバイス 7 1 などの他の制御要素によって測定され、感知され、または計算されるデータの無線伝送を使用する。図 3 の制御システム 1 0 において、新しいプロセス変数測定値または他の信号値は、特定の条件が満たされたときなどに、低速または非周期を基準にデバイス 6 0 ~ 6 4 及び 7 1 によってコントローラ 1 1 に伝送されるものとみなされる。例えば、新しいプロセス変数測定値は、デバイスによってコントローラ 1 1 に送信された最後のプロセス変数測定値に対してプロセス変数値が所定の量だけ変化したときに、コントローラ 1 1 に送信することができる。これらの信号はまた、周期的に送信することもできるが、有線プロセス制御信号などの典型的なプロセス制御システムの場合は、はるかに遅いレートであることが普通である。例えば、低速の周期的フィードバックレートは、コントローラ実行レート（コントローラが、制御信号を作成する際に使用するのための新しい制御信号を発生させるレート）よりも低くすることができ、また、本明細書で説明される制御技法を使用して、制御されているプロセス動力学の応答レートよりも 2 ~ 4 倍低いレートなどの、プロセス応答レートまたは応答時間よりも低い、それと同程度の、またはそれに類似するレートとすることができる。ここでは、プロセス応答レートは、プロセス応答時間の逆数である。当然、周期的または非周期的様式でプロセス変数測定値をいつ送信するのかを決定する他の方法を、同様にまたは代わりに実施することができる。

10

#### 【 0 0 3 0 】

理解されるように、図 3 の送信機 6 0 ~ 6 4 は、1 つ以上の制御ループ若しくは制御ルーチンで使用するために、または監視ルーチンで使用するために、それぞれのプロセス変数（例えば、流量、圧力、温度、またはレベル信号）を示す信号をコントローラ 1 1 に伝送することができる。フィールドデバイス 7 1 などの他の無線デバイスは、プロセス制御信号を無線で受信することができ、及び / または任意の他のプロセスパラメータを示す他の信号を伝送するように構成することができる。全般的に言えば、図 3 に例示されるように、コントローラ 1 1 は、入来信号を処理するプロセッサ上で実行する通信スタック 8 0 と、入来信号が測定値更新を含むときに検出するためにプロセッサ上で実行するモジュールまたはルーチン 8 2 と、測定値更新に基づいて制御を行うためにプロセッサ上で実行する 1 つ以上の制御モジュール 8 4 とを含む。検出ルーチン 8 2 は、通信スタック 8 0 を介して提供されるデータが新しいプロセス変数測定値または他のタイプの更新を含むことを示すために、フラグまたは他の信号を発生させることができる。次いで、下で更に詳細に説明されるように、新しいデータ及び更新フラグは、（機能ブロックとすることができる）制御モジュール 8 4 のうちの 1 つ以上に提供することができ、次いで、該制御モジュールが、所定の周期的実行レートでコントローラ 1 1 によって実行される。代替として、または加えて、新しいデータ及び更新フラグは、コントローラ 1 1 でまたは制御システム 1 0 の他の部分で実行される 1 つ以上の監視モジュールまたはアプリケーションに提供することができる。

20

30

#### 【 0 0 3 1 】

図 3 の無線（または他の）送信機は、全般的に、結果として低速または周期的になり、フィールドデバイス 6 0 ~ 6 4 及び 7 1 とコントローラ 1 1 との間の不規則な、または別様には頻度の低いデータ伝送を含む。しかしながら、上で述べられるように、フィールドデバイス 1 5 ~ 2 2 からコントローラ 1 1 への測定値の通信は、因習的に、コントローラの実行レートよりもはるかに速いレートで、またはプロセスの動的レートよりも少なくともはるかに速いレートで、すなわち、（制御されているプロセスの現象について）プロセス応答時間の逆数で、周期的な様式で行われるように構築されている。その結果、コントローラ 1 1 の制御ルーチンは、全般的に、コントローラ 1 1 のフィードバックループで使用するプロセス変数測定値の周期的更新のために設計される。

40

#### 【 0 0 3 2 】

例えばフィールドデバイスのうちのいくつかとコントローラ 1 1 との間の無線通信によって導入される、低速な、非周期的な、または別様には利用できない測定値更新（及び他の利用できない通信伝送）に適應するために、コントローラ 1 1 の制御及び監視ルーチン

50

(複数可)は、下で説明されるように、非周期的または非周期的または間欠的更新を含む低速を使用するときに、特に、これらの更新がコントローラ 11 の実行レートよりも低い頻度で起こるときに、更には、そのような更新をプロセス応答レート(例えば、制御されているプロセス変数のプロセス応答時間の逆数)に類似するレート(例えば、2 ~ 4 倍低いまたは同程度のレート等)で受信するときに、プロセス制御システム 10 が適切に機能することを可能にするように再構築または修正することができる。図 4 ~ 図 10 には、低速な及び/または非周期的制御に関連する通信を使用して動作するように構成される、例示的な制御スキームが更に詳細に例示される。例えば、図 4 A は、プロセス 101 に連結される位置タイプのプロセスコントローラ 100 を概略的に例示する。コントローラ 100 (図 1 及び図 3 のコントローラ 11、または、例えば図 3 の無線フィールドデバイスのうちの 1 つなどのフィールドデバイスの制御要素とすることができる)によって実施される制御スキームは、全般的に、図 3 に関連して例示され、説明される通信スタック 80 の機能性と、更新検出モジュール 82 と、制御モジュール 84 のうちの 1 つ以上とを含み、また、制御デバイスの移動可能な要素が移動するべき位置を示す制御信号を生成する。

#### 【0033】

図 4 A の例示的なシステムにおいて、コントローラ 100 は、例えばワークステーション 13 (図 1 及び図 3) のうちの 1 つから、またはプロセス制御システム 10 内の、若しくは該プロセス制御システムと通信している任意の他のソースから設定点信号を受信し、コントローラ 100 の出力からプロセス 101 に提供される 1 つ以上の制御信号 105 を発生させるように動作する。制御信号 105 を受信する他に、プロセス 101 は、矢印 104 によって概略的に示される、測定された、または測定されていない外乱を受ける場合がある。プロセス制御アプリケーションのタイプに応じて、設定点信号は、ユーザ、調整ルーチンなどによるなどして、プロセス 101 の制御中の任意のときに変化させることができる。当然、プロセス制御信号 105 は、弁と関連付けられるアクチュエータを制御することができ、またはプロセス 101 の動作の応答に影響を及ぼすように任意の他のフィールドデバイスを制御することができる。プロセス制御信号 105 の変化に対するプロセス 101 の応答は、送信機、センサ、または他のフィールドデバイス 106 によって測定または検知され、例えば、図 3 に例示される送信機 60 ~ 64 のうちの任意の 1 つに対応することができる。送信機 106 とコントローラ 100 との間の通信リンクは、無線接続を含むことができ、破線を使用して図 4 A に例示される。

#### 【0034】

簡単な実施形態において、コントローラ 100 は、PID 制御ルーチンの 1 つの形態である、PI 制御ルーチンなどの、単一入力、単一出力の閉ループ制御ルーチンを実施することができる。故に、コントローラ 100 は、通信スタック 80 を含むいくつかの標準 PI コントローラ要素と、加算ブロック 108、比例ゲイン要素 110、更なる加算ブロック 112、及び高低リミッタ 114 を含む制御信号生成部とを含む。制御ルーチン 100 はまた、フィルタ 116 と、選択ブロック 118 を含む積分出力スイッチとを含む、直接フィードバック経路も含む。フィルタ 116 は、高低リミッタ 114 の出力に連結され、スイッチのブロック 118 は、フィルタ 116 の出力に連結され、コントローラ 100 によって発生させている制御信号の積分またはリセット寄与または成分を加算ブロック 112 に提供する。

#### 【0035】

コントローラ 100 の動作中に、加算ブロック 108 は、誤差信号を生成するために、設定点信号と、コントローラ 100 内の通信スタック 80 から提供される直前に受信したプロセス変数測定値とを比較する。比例ゲイン要素 110 は、制御信号の比例寄与または成分を生成するために、例えば、誤差信号に比例ゲイン値  $K_p$  を乗じることによって、誤差信号に対して動作する。次いで、加算ブロック 112 は、本質的に制限されない制御信号を生成するために、ゲイン要素 110 の出力(すなわち、比例寄与)と、積分若しくはリセット寄与、またはフィードバック経路によって生成される制御信号の成分とを組み合わせる。次いで、リミッタブロック 114 は、プロセス 101 を制御するために送信され

10

20

30

40

50

る制御信号 105 を生成するために、加算ブロック 112 の出力に対して高低制限を行う。  
【0036】

重要なことに、コントローラ 100 のフィードバック経路内のフィルタ 116 及びブロックまたはスイッチ 118 は、制御信号の積分またはリセット寄与成分を生成するために、以下の様式で動作する。リミッタ 114 の出力を受信するように連結されるフィルタ 116 は、リミッタ 114 の出力値及び制御アルゴリズム 100 の実行期間または時間に基づいて、制御信号 105 に対する予想されるプロセス応答の指示を生成する。フィルタ 116 は、この予想されるプロセス応答信号をスイッチまたはブロック 118 に提供する。スイッチまたはブロック 118 は、新しいプロセス変数測定値を受信したときには常に、スイッチまたはブロック 118 の出力でフィルタ 116 の出力をサンプリング及びクランプし、通信スタック 80 で次のプロセス変数出力を受信するまで、この値を維持する。したがって、スイッチ 118 の出力は、最後の測定値更新でサンプリングしたフィルタ 116 の出力のままである。

10

【0037】

フィルタ 116 によって生成されるような、加算器 108 の出力の変化に対する予想されるプロセス応答は、下でより詳細に説明されるように、一次モデルを使用して近似することができる。しかしながら、より全般的には、予想されるプロセス応答は、プロセス 101 の任意の適切なモデルを使用して生成することができ、コントローラ 100 のフィードバック経路に組み込まれるモデルに、または制御信号に対する積分若しくはリセット寄与を決定することと関連付けられるフィルタまたはモデルに限定されない。例えば、予想されるプロセス応答を提供するためにモデルを利用するコントローラは、制御ルーチン 100 が PID 制御スキームを実行するように、微分寄与を組み込むことができる。例示的なタイプの微分寄与を組み込むいくつかの実施例は、図 6 ~ 図 8 に関連して下で説明される。

20

【0038】

図 4 A のフィルタ 116 の動作をより詳細に論じる前に、積分またはリセット寄与を決定するために、従来の PI コントローラを、ポジティブフィードバックネットワークを使用して実行することができることに留意することが有用である。数学的に、従来の PI を実施するための伝達関数は、制約されない制御に対する、すなわち出力が制限されない場合の標準的な式に等しいことを示すことが分かる。特に：

30

【数 1】

$$\frac{O(s)}{E(s)} = K_P \left( 1 + \frac{1}{sT_{Reset}} \right)$$

式中、 $K_P$  - 比例ゲイン

$T_{Reset}$  = リセット、秒

$O(s)$  = 制御出力

$E(s)$  = 制御誤差

40

図 4 A に示されるように、コントローラ 100 内でポジティブフィードバック経路を使用する 1 つの利点は、コントローラ出力が、高くまたは低く、すなわちリミッタ 114 によって制限されるときに、リセット寄与をワインドアップすることが自動的に防止されることである。

【0039】

いずれにしても、下で説明される制御技法は、コントローラがプロセス変数の非周期的更新を受信したときに、リセットまたは積分寄与を決定するためのポジティブフィードバ

50

ック経路を使用することを可能にするが、それでも、新しいプロセス変数測定値の受信の間に起こる設定点変化またはフィードフォワード変化が生じた場合に、ロバストなコントローラ応答を可能にする。具体的には、ロバストな設定点変化をコントローラの動作に提供するために、フィルタ 1 1 6 は、フィルタのこの出力が加算ブロック 1 1 2 に提供されたかどうかにかかわらず、コントローラ 1 0 0 のそれぞれまたは全ての実行の間、予想されるプロセス応答の新しい指示または値を算出するように構成される。その結果、コントローラ 1 0 0 が通信スタック 8 0 からの新しいプロセス測定値更新を受信した直後に発生させたフィルタ 1 1 6 の出力だけが、加算器 1 1 2 において積分またはリセット寄与として使用される場合であっても、コントローラルーチンの各実行サイクル中に、フィルタ 1 1 6 の出力が新しく再生される。

10

【 0 0 4 0 】

具体的には、フィルタ 1 1 6 によって生成されるような予想される応答の新しい指示は、現在のコントローラ出力（すなわち、リミッタ 1 1 4 の後の制御信号）、最後の（すなわち、直前の）コントローラの実行サイクル中に生成されるフィルタ 1 1 6 によって生成される予想される応答の指示、及びコントローラの実行期間から、各コントローラ実行サイクル中に算出される。その結果、各コントローラ実行サイクル中に新しいプロセス応答の推定を生成するために実行されるので、フィルタ 1 1 6 は、連続的に更新されるように本明細書で説明される。各コントローラ実行サイクル中に新しい予想されるプロセス応答またはフィルタを生成するために、連続的に更新されるフィルタ 1 1 6 によって実施することができる、例示的な式が下に記載される。

20

【 数 2 】

$$F_N = F_{N-1} + (O_{N-1} - F_{N-1}) * \left(1 - e^{\frac{-\Delta T}{T_{Reset}}}\right)$$

式中、 $F_N$  - 新しいフィルタ出力

$F_{N-1}$  = フィルタ出力の最後の実行

30

$O_{N-1}$  = コントローラ出力の最後の実行

$\Delta T$  = コントローラ実行期間

【 0 0 4 1 】

ここで、新しいフィルタ出力  $F_N$  は、直前のフィルタ出力  $F_{N-1}$ （すなわち、現在のフィルタ出力値）に、現在のコントローラ出力値  $O_{N-1}$  と現在のフィルタ出力値  $F_{N-1}$  との間の差分にリセット時間  $T_{Reset}$  及びコントローラの実行期間  $T$  に依存する係数を乗じたものを加えて決定される減衰成分を加えて、繰り返し決定されることに気が付くであろう。この様式で連続的に更新するフィルタを使用することで、制御ルーチン 1 0 0 は、新しいプロセス変数測定値を受信したときに積分制御信号入力を算出したときに、予想されるプロセス応答をより良好に決定することができ、それによって、設定点の変化に対して、または 2 つのプロセス変数測定値の受信の間に起こる他のフィードフォワード外乱に対してより敏感になる。より具体的には、設定点の変化（新しいプロセス測定値の受信を伴わない）が、加算器 1 0 8 の出力での誤差信号の変化を直ちにもたらし、これは、制御信号の比例寄与成分を変化させ、したがって、制御信号を変化させることに気が付くであろう。その結果、フィルタ 1 1 6 は、変化した制御信号に対するプロセスの新しい予想

40

50

される応答を直ちに生成し始め、したがって、コントローラ 100 が新しいプロセス測定値を受信する前にその出力を更新する。次いで、コントローラ 100 が新しいプロセス測定値を受信し、フィルタ出力のサンプルが、制御信号の積分またはリセット寄与成分として使用されるように、スイッチ 118 によって加算器 112 の入力にクランプされたときに、フィルタ 116 は、少なくともある程度は、設定点の変化に対するプロセス 101 の応答に反応した、またはそれを組み込んだ、予想されるプロセス応答を繰り返している。

#### 【0042】

過去において、米国特許第 7,587,252 号及び第 7,620,460 号で説明されるシステムのように、非周期的に更新されるコントローラのフィードバック経路において使用されるリセット寄与フィルタは、新しいプロセス変数測定値を使用することができたときに、予想される応答の新しい指示だけを算出した。その結果、リセット寄与フィルタは、設定点変化またはフィードフォワード外乱が任意の測定値更新から完全に独立していたので、プロセス変数測定値の受信の間に起こった設定点変化またはフィードフォワード外乱を補償しなかった。例えば、設定点変化またはフィードフォワード外乱が 2 つの測定値更新の間に起こった場合は、予想される応答の新しい指示の算出が最後の測定値更新以来の時間及び現在のコントローラ出力 105 に基づいていたので、コントローラの予想されるプロセス応答が歪められる可能性があった。その結果、フィルタ 116 は、コントローラでの 2 つのプロセス変数測定値の受信の間に起こった設定点変化（または他のフィードフォワード外乱）から生じた、プロセス（または制御信号）の時間変化を補償し始めることができなかった。

#### 【0043】

しかしながら、理解されるように、図 4 A の制御ルーチン 100 は、非周期的測定値の算出に基づくことによって、予想されるプロセス応答を提供し、一方で、加えて、設定点の変化（またはコントローラ 100 へのフィードフォワード入力として使用される任意の測定された外乱）によって引き起こされる変化を補償するために、2 つの測定値の受信の間の予想される応答を決定する。したがって、上で説明される制御技法は、予想されるプロセス応答に影響を及ぼし得る設定点変化、測定される外乱に対するフィードフォワードアクションなどに適応することができ、したがって、よりロバストな制御応答を提供する。

#### 【0044】

理解されるように、図 4 A に例示される制御技法は、制御ブロックまたはルーチン 100 の各実行について連続的に更新されるフィルタ 116（例えば、リセット寄与フィルタ）を介して、予想される応答の指示を算出する。ここで、コントローラ 100 は、制御ブロックの各実行について、予想される応答の新しい指示を算出するように、連続的に更新されるフィルタ 116 を構成する。しかしながら、フィルタ 116 の出力を加算ブロック 112 への入力として使用するべきであるかどうかを判定するために、通信スタック 80 及び、いくつかの実施例において、更新検出モジュール 82（図 3）は、新しいプロセス変数測定値を受信したときに、積分出力スイッチ 118 の新しい値フラグを発生させるために、送信機 106 からの入来データを処理する。この新しい値フラグは、加算器 112 の入力に対するこのコントローラ繰り返しについてフィルタ出力値をサンプリングし、クランプすることをスイッチ 118 に通知する。

#### 【0045】

新しい値フラグが通信されるかどうかにかかわらず、連続的に更新されるフィルタ 116 は、制御ルーチンの各繰り返しについて、予想される応答の指示を算出し続ける。予想される応答のこの新しい指示は、制御ブロックの各実行について、積分出力スイッチ 118 に搬送される。新しい値フラグの存在に応じて、積分出力スイッチ 118 は、連続的に更新されるフィルタ 116 からの予想される応答の新しい指示を加算ブロック 112 に通過させることを可能にすることと、制御ブロックの最後の実行中に加算ブロック 112 に以前に送達された信号を維持することとの間で切り替える。より具体的には、新しい値フラグが通信されるときに、積分出力スイッチ 118 は、連続的に更新されるフィルタ 116 からの予想される応答の直前に算出された指示を、加算ブロック 112 に渡すことを可

10

20

30

40

50



能にする。その逆に、新しい値フラグが存在しない場合、積分出力スイッチ 118 は、最後の制御ブロック繰り返しからの予想される応答の指示を加算ブロック 112 に再送信する。換言すれば、積分出力スイッチ 118 は、新しい値フラグがスタック 80 から通信される度に、予想される応答の新しい指示にクランプするが、新しい値フラグが存在しない場合は、いかなる予想される応答の新しく算出された指示も加算ブロック 112 に到達させないようにする。

#### 【0046】

この制御技法は、新しい測定値が通信されるかどうかにかかわらず、連続的に更新されるフィルタ 116 が、予想されるプロセス応答をモデル化し続けることを可能にする。制御出力が設定点変化または測定された外乱に基づくフィードフォワードアクションの結果として変化する場合、新しい値フラグの存在にかかわらず、連続的に更新されるフィルタ 116 は、各制御ルーチン繰り返しにおいて、予想される応答の新しい指示を算出することによって、予想されるプロセス応答を正しく反映する。しかしながら、予想される応答の新しい指示（すなわち、リセット寄与または積分成分）は、新しい値フラグが（積分出力スイッチ 118 を介して）通信されたときに、コントローラの算出にだけ組み込まれる。

#### 【0047】

図 4 B に例示されるグラフ 200 は、コントローラ 100 がいくつかの設定点変化に応答したときに、プロセス出力信号 202 を定常状態値まで駆動する際の、図 4 A のコントローラ 100 のシミュレーションされた動作を描写する。図 4 B では、プロセス制御システムの無線動作中の、プロセス出力信号 202（太線で例示される）が、設定値信号 204（細線で示される）と比較して示される。グラフ 200 の下部の時間軸に沿って矢印によって示されるように、設定点変化が起こったときに、コントローラ 102 は、プロセス出力を駆動する制御信号を生成することによって、新しい設定値（すなわち、定常状態値）に反応するように応答する。例えば、図 4 B に例示されるように、設定点変化は、時間  $T_1$  で起こるが、これは、設定値信号 204 の大きさがより高い値からより低い値まで大幅に変化することから明らかである。それに応じて、コントローラ 102 は、時間  $T_1$  と時間  $T_2$  との間で出力信号 202 が呈するような滑らかな過渡曲線において、新しい定常状態に対する設定点または設定点値と関連付けられるプロセス変数を駆動する。同様に、図 4 B において、第 2 の設定点変化は、時間  $T_2$  で起こるが、これは、設定値信号 204 の大きさがより低い値からより高い値まで大幅に変化することから明らかである。それに応じて、コントローラ 102 は、時間  $T_2$  と  $T_3$  との間で出力信号 202 によって示されるような滑らかな過渡曲線において、新しい定常状態に対する設定点または設定点値と関連付けられるプロセス変数を制御する。その結果、図 4 B から分かるように、上で説明される制御ルーチンを実施するコントローラ 100 は、ロバストな様式で、非周期的無線制御システムにおける設定点変化の補償を可能にする。フィードフォワード外乱は、制御アクション動作において測定され、含めることができるので、上で説明される制御ルーチンを実施するコントローラ 100 はまた、非周期的無線制御システムにおける制御出力のフィードフォワード変化の補償も可能にすることができる。

#### 【0048】

図 4 A の簡単な P I コントローラ構成は、フィルタ 116 の出力を制御信号に対するリセット寄与として直接使用し、この事例において、閉ループ制御ルーチンのリセット寄与（例えば、上で提示される、連続的に更新されるフィルタの式）は、プロセスが定常状態動作を呈するかどうかを判定する際に、プロセス応答の正確な表現を提供することができることに留意されたい。しかしながら、むだ時間が支配的なプロセスなどの他のプロセスは、予想されるプロセス応答をモデル化するために、追加的な構成要素を図 4 A のコントローラに組み込むことを必要とする場合がある。一次モデルによって十分に表されるプロセスに関しては、全般的に、P I（または P I D）コントローラのリセット時間を決定するために、プロセス時定数を使用することができる。より具体的には、リセット時間がプロセス時定数に等しく設定された場合、リセット寄与は、全般的に、経時的に制御ルーチ

10

20

30

40

50

ン 1 0 0 が予想されるプロセス応答を反映するように、比例寄与を無効にする。図 4 A に例示される実施例において、リセット寄与は、プロセス時定数と同じ時定数を有するフィルタを有するポジティブフィードバックネットワークによって達成することができる。他のモデルを使用することができるが、ポジティブフィードバックネットワーク、フィルタ、またはモデルは、既知または近似のプロセス時定数を有するプロセスの予想される応答を決定するための好都合な機構を提供する。P I D 制御を必要とするプロセスの場合、P I D 出力に対する、レートとしても知られる微分寄与は、新しい測定値を受信したときにだけ再計算し、更新することもできる。そのような事例において、微分算出は、最後の新しい測定以来の経過時間を使用することができる。プロセス測定値の非周期的受信を使用してより複雑なプロセスを制御するために他のコントローラ成分を使用することができるが、設定点変化に応答してロバストな制御を提供するために図 4 A のフィルタリング技法を使用することができる、コントローラのいくつかの実施例が、図 5 ~ 図 8 と併せて下で説明される。

10

#### 【 0 0 4 9 】

以下、図 5 を参照すると、上で説明されるように、制御技法に従って構成される、代替のコントローラ（または制御要素）1 2 0 は、図 4 A に例示されるコントローラ 1 0 0 に多くの点で類似する。その結果、双方のコントローラに共通な要素は、同じ参照番号で識別される。しかしながら、コントローラ 1 2 0 は、追加的な要素を制御ルーチンに組み込み、該ルーチンは、測定値の伝送の間の予想されるプロセス応答を決定する。この事例において、プロセス 1 0 1 は、相当な量のむだ時間を有することを特徴とすることができ、その結果、むだ時間ユニットまたはブロック 1 2 2 が、むだ時間補正のためのコントローラモデルに含まれる。むだ時間ユニット 1 2 2 の組み込みは、全般的に、プロセス応答のより正確な表現に到達するのを補助する。より具体的には、むだ時間ユニット 1 2 2 は、任意の所望の様式で実施することができ、スミス予測手段または他の既知の制御ルーチンに共通な方法を含むか、または利用することができる。しかしながら、この状況において、連続的に更新されるフィルタ 1 1 6 及びスイッチモジュール 1 1 8 は、設定点変化に応答してロバストな制御を提供するために、図 4 A のコントローラ 1 0 0 に関して上で説明されるものと同じ様式で動作する。

20

#### 【 0 0 5 0 】

図 6 は、別の代替のコントローラ（または制御要素）1 3 0 を描写するが、微分またはレート寄与成分がコントローラ 1 3 0 に組み込まれる点で、図 4 A において上で説明されるコントローラ 1 0 0 と異なる。微分寄与を組み込むことによって、コントローラ 1 3 0 によって実施される制御ルーチンは、いくつかの事例において、比例 - 積分 - 微分（P I D）制御スキームが実施されるように、追加的なフィードバック機構を含む。

30

#### 【 0 0 5 1 】

図 6 の制御ルーチンまたは技法は、図 4 A の積分寄与と関連して上で説明される様式に類似する様式で構成される微分寄与を含み、プロセス測定値の非周期的な、または別様には利用できない更新に適應する。微分寄与は、最後の測定値更新以来の経過時間に基づいて再構成することができる。この様式で、微分寄与（及び結果として生じる出力信号）のスパイクが回避される。より具体的には、図 6 の微分寄与は、比例及び積分寄与専用の要素と並行に加算ブロック 1 0 8 から誤差信号を受信する、微分ブロック 1 3 2 によって決定される。他の P I D 構成（例えば、シリアル構成）も利用することができるが、比例、積分、及び微分寄与は、図 6 に示されるように、加算ブロック 1 3 4 において組み合わせられる。

40

#### 【 0 0 5 2 】

信頼できない伝送及び、より全般的には、測定値更新の利用不能に適應するために、微分寄与は、通信スタック 8 0 からの新しい値フラグによって示されるように、測定値更新を受信するまで、最後に決定された値に維持される。この技法は、制御ルーチンが、制御ルーチンの通常の、または確立された実行レートに従って周期的実行を続けることを可能にする。更新された測定値を受信すると、図 6 に例示されるように、微分ブロック 1 3 2

50

は、以下の式に従って微分寄与を決定することができる。

【数 3】

$$O_D = K_D \cdot \frac{e_N - e_{N-1}}{\Delta T}$$

式中、 $e_N$  = 現在の誤差

$e_{N-1}$  = 最後の誤差

$\Delta T$  = 新しい値を通信して以来の経過時間

$O_D$  = コントローラの微分項

$K_D$  = 微分ゲイン係数

10

【0053】

微分寄与を決定するためのこの技法によって、プロセス変数（すなわち、制御入力）に対する測定値更新を、出力スパイクを生成することなく、1つ以上の実行期間の間にわたって喪失することができる。通信が再確立されたときに、微分寄与式の項（ $e_N - e_{N-1}$ ）は、微分寄与の標準的な算出において発生させた値と同じ値を発生させることができる。しかしながら、標準的なPID技法の場合、微分寄与を決定する際の除数は、実行期間である。対照的に、本制御技法は、2つの成功裏に受信した測定値の間の経過時間を利用する。実行期間よりも長い経過時間を使用することで、制御技法は、標準PID技法よりも小さい微分寄与を生成し、また、スパイクが低減される。

20

【0054】

経過時間の決定を容易にするために、通信スタック80は、図6に示されるように、上で説明される新しい値フラグを微分ブロック132に提供することができる。代替の実施例は、その値に基づいて、新しい測定値または更新の検出を含むこと、または関与させることができる。また、プロセス測定値を、比例または微分成分の算出において誤差の代わりに使用することができる。より全般的には、通信スタック80は、プロセス101内の任意のフィールドデバイス、コントローラの外部のプロセス制御要素などを含む、プロセス101との通信インターフェースを実施するために、任意のソフトウェア、ハードウェア、またはファームウェア（またはそれらの任意の組み合わせ）を含むこと、または関与させることができる。しかしながら、図6のコントローラ130において、連続的に更新されるフィルタ116及びスイッチモジュール118は、設定点変化に応答してロバストな制御を提供するために、図4Aのコントローラ100に関して上で説明されるものと同じように動作する。

30

【0055】

図3、図4A、及び図5～図6に関連して説明される、コントローラによって制御されるアクチュエータまたは他の下流の要素は、特に、コントローラまたは制御要素と下流のアクチュエータまたは他の要素との間にいかなる通信もない期間の後に、それでも、突然の変化を伴う制御信号を受信する場合がある。結果として生じる制御アクションは、いくつかの事例において、プラントの動作に影響を与えるのに十分突然である場合があり、そのような突然の変化は、不適切なレベルの不安定性につながる場合がある。

40

【0056】

コントローラと下流の要素との間の通信の喪失による突然の制御変化の可能性は、制御信号に対するフィードバック寄与（複数可）を決定するときに、最後の実行期間中に、コントローラ出力の代わりに実際の下流のデータを組み込むことによって対処することができる。全般的に言えば、そのような実際の下流のデータは、制御信号に対する応答のフィードバック指示を提供し、したがって、制御信号を受信する下流の要素（例えば、プロセス制御モジュール）またはデバイス（例えば、アクチュエータ）によって測定または算出することができる。そのようなデータは、最後の実行からのコントローラ出力などの、制

50

御信号に対する暗黙の応答の代わりに提供される。図 4 A 及び図 5 ~ 図 6 に示されるように、連続的に更新されるフィルタ 1 1 6 は、下流の応答の暗黙の指示として制御信号 1 0 5 を受信する。そのような暗黙のデータの使用は、アクチュエータなどの下流の要素が制御信号の通信を受信したことを効果的に想定し、したがって、制御信号に適切に応答している。実際のフィードバックデータもまた、制御されているプロセス変数の測定値などの他の応答指示と異なる。

#### 【 0 0 5 7 】

図 7 は、制御信号に응答する下流のデバイスまたは要素からのアクチュエータ位置データを受信する、例示的なコントローラ 1 4 0 を描写する。下流のデバイスまたは要素は、しばしば、アクチュエータに対応して、アクチュエータ位置の測定値を提供する。より一般的には、下流のデバイスまたは要素は、P I D 制御ブロック、制御セクタ、スプリッタ、または制御信号によって制御される任意の他のデバイス若しくは要素に対応するか、またはそれを含むことができる。示される例示的な事例において、アクチュエータ位置データは、制御信号に対する応答の指示として提供される。したがって、アクチュエータ位置データは、プロセス変数の測定値更新の欠如にもかかわらず、制御ルーチンの連続的な実行期間中に、コントローラ 1 4 0 によって利用される。この目的のために、連続的に更新されるフィルタ 1 1 6 は、入来フィードバックデータに対するインターフェースを確立する通信スタック 1 4 6 を介して、アクチュエータ位置データを受信することができる。この例示的な事例において、フィードバックデータは、制御信号に対する応答の 2 つの指示、アクチュエータ位置、及びプロセス変数を含む。

#### 【 0 0 5 8 】

前の実施例と同様に、連続的に更新されるフィルタ 1 1 6 は、プロセス変数に対する測定値更新の欠如に関与する状況に適応するように構成される。連続的に更新されるフィルタ 1 1 6 は、新しい測定値フラグを受信した後に発生させたフィルタ出力だけが加算器 1 1 2 で使用されるという事実にもかかわらず、そのような欠如の間、その出力を同様に再算出する。しかしながら、測定値更新を受信すると、連続的に更新されるフィルタ 1 1 6 は、その出力を修正するために、もはや制御信号のフィードバックに依存しない。むしろ、アクチュエータからの実際の応答データは、以下に示されるように利用される。

#### 【 数 4 】

$$F_N = F_{N-1} + (A_{N-1} - F_{N-1}) * \left(1 - e^{\frac{-\Delta T}{T_{Reset}}}\right)$$

式中、 $F_N$  - 新しいフィルタ出力

$F_{N-1}$  = フィルタ出力の最後の実行

$A_{N-1}$  = コントローラ出力の最後の実行

$\Delta T$  = コントローラ実行期間

$T_{Reset}$  = リセット時間

#### 【 0 0 5 9 】

制御信号に対する応答の実際の指示の使用は、周期的通信の期間中、及び P I D 制御要素から下流の要素、例えばアクチュエータへの非周期的通信の期間または通信喪失の後の

双方で、制御技法の正確さを向上させるのを補助することができる。しかしながら、実際の応答指示の伝送は、典型的に、異なるデバイスにおいて実施される場合に、フィールドデバイスとコントローラとの間に追加的な通信を必要とする。そのような通信は、上で説明されるように、無線とすることができ、したがって、信頼できない伝送または電力制限の影響を受けやすくなる場合がある。他の理由もまた、フィードバックデータの利用不能につながる場合がある。

#### 【 0 0 6 0 】

上で説明されるように、本明細書で論じられる制御技法はまた、そのような応答指示が周期的またはタイムリーな様式で通信されない状況に対処することもできる。すなわち、本制御技法の適用は、プロセス変数の測定値更新の欠如に限定する必要はない。むしろ、本制御技法は、アクチュエータの位置または下流の制御要素の出力などの他の応答指示の欠如に関与する状況に対処するように、好都合に利用することができる。なお更に、本制御技法は、コントローラ（または制御要素）から、フィールドデバイス（例えば、アクチュエータ）または別の制御要素（例えば、カスケード型 P I D 制御、スプリッタなど）などの下流の要素への伝送の喪失、遅延、または他の利用不能に関与する状況に対処するために利用することができる。

10

#### 【 0 0 6 1 】

コントローラ若しくは制御要素への追加的なデータ（すなわち、応答指示または下流の要素のフィードバック）の、またはコントローラ若しくは制御要素からの追加データ（すなわち、制御信号）の、無線または他の信頼できない伝送は、通信の問題及び／または課題に対する追加的な可能性を提供する。上で説明されるように、下流の要素（例えば、アクチュエータ）からのフィードバックは、積分寄与（または他の制御パラメータ若しくは寄与）を決定する際に関与させることができる。この実施例において、制御ルーチンは、上で説明される実施例においてフィードバックされる単一プロセス変数ではなく、2つのフィードバック信号に依存する。更に、制御信号が下流の要素に今まで到達していない場合、プロセスは、制御スキームの恩恵を受けない。これらの信号のうちのいずれか1つの伝送は、遅延するか、または喪失する場合があり、したがって、本明細書で説明される技法は、どちらの可能性にも対処する。

20

#### 【 0 0 6 2 】

フィルタまたは他の制御計算に関与する応答指示の欠如は、更新を受信するまで、予想される応答（または他の制御信号成分）の指示を維持することによって対処することができる。

30

#### 【 0 0 6 3 】

制御信号が下流の要素に到達しないときに、下流の要素からの応答指示（すなわち、フィードバック）は、変化しない。そのような事例において、値の変化の欠如は、値の変化を受信するまで、予想される応答（または他の制御信号成分）の指示を同様に維持するために、コントローラ（または制御要素）の論理をトリガーすることができる。

#### 【 0 0 6 4 】

本制御技法はまた、実際のフィードバックデータが所望されないか、または利用できない状況でも実施することができる。前者の事例は、制御信号に対する暗黙の応答を使用するといった簡単さが有益である状況において好都合である場合がある。例えば、実際のフィードバックデータの通信は、問題を含むか、または非実用的である場合がある。後者の事例は、上で説明されるように、位置測定値データを提供するように構成されないアクチュエータまたは他のデバイスが関与する場合がある。より古いデバイスは、そのような性能を有しない場合がある。

40

#### 【 0 0 6 5 】

そのようなデバイスに適應するために、暗黙または実際のいずれかの応答指示を本制御技法が使用することを可能にするように、スイッチまたは他のデバイスを提供することができる。図 8 に例示されるように、コントローラ 150 は、スイッチ 152 に連結され、よって、暗黙及び実際の応答指示の双方を受信する。この事例において、制御スキームの

50

実施が応答指示のタイプを知ること依存しないので、コントローラ 150 は、上で説明されるコントローラのうちのいずれかと同一とすることができる。スイッチ 152 は、ソフトウェア、ハードウェア、ファームウェア、またはそれらの任意の組み合わせで実施することができる。スイッチ 152 の制御は、コントローラ 150 及び任意の制御ルーチンの実施とは無関係とすることができる。代替的にまたは追加的に、コントローラ 150 は、スイッチ 152 を構成するために、制御信号を提供することができる。更に、スイッチ 152 は、コントローラ自体の一部として実施することができ、また、いくつかの事例では、通信スタックの一部またはコントローラの他の部分として統合することができる。

#### 【0066】

制御方法、システム、及び技法の実践は、いずれか 1 つの特定の無線アーキテクチャまたは通信プロトコルに限定されない。適切な例示的なアーキテクチャ及び通信サポートスキームは、2005 年 6 月 17 日に出願された米国特許出願第 11/156,215 号、名称「Wireless Architecture and Support for Process Control Systems」で説明されており、その開示全体が参照により本明細書に組み込まれる。実際に、制御ルーチンに対する修正は、制御ルーチンが周期的な方法で実施されるが、各制御繰り返しに対するプロセス変数測定値を伴わない、任意の状況に十分適している。他の例示的な状況としては、サンプリングされた値が、例えばアナライザによって、または実験室サンプルを介して、不規則に提供されるか、または減多に提供されない場合が挙げられる。

#### 【0067】

本制御技法の履行は、単一入力、単一出力の PID 制御ルーチン（PI 及び PD ルーチンを含む）とともに使用することに限定されず、むしろ、多数の異なる多重入力及び/または多重出力制御スキーム、並びにカスケード型制御スキームに適用することができる。より全般的には、本制御技法はまた、1 つ以上のプロセス変数、1 つ以上のプロセス入力、またはモデル予測制御（MPC）などの他の制御信号に関与する、任意の閉ループモデルに基づく制御ルーチンの状況に適用することもできる。

#### 【0068】

図 9 は、本明細書で説明される原理を使用するが、速度に基づくコントローラの形態でコントローラ 300 を有するように構成される、更なる例示的な制御システムを例示する。図 9 の例示的なシステムにおいて、コントローラ 300 は、例えばワークステーション 13（図 1 及び図 3）のうちの 1 つから、またはプロセス制御システム内の、若しくは該プロセス制御システムと通信している任意の他のソースから設定点信号を受信し、コントローラ 300 の出力からプロセス 301 に提供される、1 つ以上の制御信号 305 を発生させるように動作する。制御信号 305 を受信する他に、プロセス 301 は、図 4 において矢印 304 によって概略的に示される、測定された、または測定されていない外乱を受ける場合がある。プロセス制御アプリケーションのタイプに応じて、設定点信号は、ユーザ、調整ルーチンなどによるなどして、プロセス 301 の制御中の任意のときに変化させることができる。当然、プロセス制御信号 305 は、弁と関連付けられるアクチュエータを制御することができ、またはプロセス 301 の動作の応答に影響を及ぼすように任意の他のフィールドデバイスを制御することができる。プロセス制御信号 305 の変化に対するプロセス 301 の応答は、送信機、センサ、または他のフィールドデバイス 306 によって測定または検知され、例えば、図 3 に例示される送信機 60 ~ 64 のうちの任意の 1 つに対応することができる。送信機 306 とコントローラ 300 との間の通信リンクは、無線接続を含むことができ、破線を使用して図 9 に例示される。しかしながら、このリンクはまた、有線通信リンクまたは他のタイプの通信リンクとすることもできる。議論の目的で、送信機 306 は、低速または間欠的な更新レートで、制御されるプロセス変数（すなわち、被制御変数）または制御されたプロセス変数と相関されるプロキシ変数を測定しているものとみなされる。この低速の更新レートは、周期的または非周期的とすることができ、また、制御されたプロセス変数と関連付けられるプロセス動力学的プロセス応答レートと同程度の大きさであるとみなされる。したがって、プロセス変数測定値は、プロセ

10

20

30

40

50

スの応答時間よりも長い時間間隔あたり一度、プロセス応答時間に類似する時間間隔あたり一度、またはプロセス応答時間よりも僅かに短い時間間隔あたり一度提供される。したがって、いくつかの事例において、この更新レートは、プロセス応答レートの  $1/2 \sim 1/4$  (プロセス応答時間の逆数) とすることができる。

#### 【0069】

図9に例示される簡単な実施形態において、コントローラ300は、PI制御ルーチンなどの単一入力、単一出力の閉ループ制御ルーチンを実施することができ、これは、PID制御ルーチンの1つの形態である。故に、コントローラ300は、通信スタック380を含むいくつかの標準PIコントローラ要素と、加算ブロック308、比例ゲイン要素310、及び更なる加算ブロック312を含む制御信号生成部とを含む。制御ルーチン300はまた、フィルタ316と、選択ブロック318を含む積分出力スイッチとを含む、直接積分フィードバック経路も含む。しかしながら、この事例において、図9のPIコントローラは、制御信号の比例及び積分成分に関する位置及び差分の制御算出を行うように構成される。したがって、コントローラ300はまた、比例成分算出経路に配置される差分ブロック320、積分成分算出経路に配置される加算器322、及び制御信号305を生成するために差分的に算出された制御成分をそこへの入力で使用するブロック324も含む。全般的に言えば、ブロック324は、コントローラ300内で発生させたときに、位置(速度)制御信号の変化をスケーリングするか、または別様には、この信号を、制御デバイスに送信されるアナログまたはデジタル信号に変換して、制御デバイスが、特定の期間にわたってある方向または別の方向にある特定の量だけ移動するように指示する。このブロック324は、例えば、パルス幅変調信号、パルス信号、ターンオン時間を示すデジタル信号、または位置の変化の大きさを示す任意の他の信号を、経時的に制御デバイスに送信することができる。

#### 【0070】

図9に例示されるように、積分フィルタ316は、加算器322の出力に連結され、次にこの出力が加算器312の出力を受信するように連結され、一方で、スイッチのブロック318は、フィルタ316の出力に連結され、コントローラ300によって発生させている制御信号の積分またはリセット寄与または成分を加算ブロック312に提供する。

#### 【0071】

コントローラ300の各繰り返しまたは動作中に、加算ブロック308は、誤差信号(e)を生成するために、設定点信号と、コントローラ300内で通信スタック380から提供される直前に受信したプロセス変数測定値とを比較する。比例ゲイン要素またはブロック310は、速度制御信号の比例寄与または成分を生成するために、例えば、誤差信号eに比例ゲイン値 $K_p$ を乗じることによって、誤差信号eに対して動作する。差分ブロック320は、次いで、(最後または直前のコントローラ繰り返し中生成される)ゲインブロック310の現在の出力とゲインブロック310の直前の値との間の差分を決定することによって、最後のコントローラ繰り返し以来の比例ゲイン値の変化を決定する。次いで、加算ブロック312は、出力ブロック324に提供される速度制御信号326を生成するために、変化ユニット320の出力(すなわち、速度に基づく比例寄与)と、積分フィードバック経路によって生成される制御信号の積分またはリセット寄与または成分とを組み合わせる。

#### 【0072】

しかしながら、重要なことに、コントローラ300の積分フィードバック経路内の加算器322、フィルタ316、及びブロックまたはスイッチ318は、制御信号の積分またはリセット寄与成分を生成するために、以下の様式で動作する。ここで、加算器322は、各コントローラ繰り返し中に、加算器312の出力(すなわち、移動可能な制御要素の位置の変化を表す、速度に基づく制御信号)を受信するように連結され、また、その値と、(コントローラ300の最後の繰り返し中に発生させた)加算器322の以前の出力Sとを合計して、それによって、実際には、ある特定の期間にわたる出力信号の変化の変化を積分または合計する。加算器322の新しい出力Sは、積分フィルタ316に提供され

10

20

30

40

50

、該積分フィルタは、図 9 において R で示される、制御信号 305 に対する予想されるプロセス応答の指示を生成する。フィルタ 316 は、この予想されるプロセス応答信号 R をスイッチまたはブロック 318 に提供する。しかしながら、図 9 に示されるように、スイッチまたはブロック 318 は、新しいプロセス変数測定値を受信したときには常に、スイッチまたはブロック 318 の出力でフィルタ 316 の出力をサンプリングし、クランプするが、別様には、いかなる新しいプロセス変数測定値も利用できない制御繰り返し中の積分制御寄与として、ゼロ (0.0) 値を加算器 312 に提供する。したがって、各コントローラ繰り返し中に新しい制御信号を発生させるために積分制御寄与として提供される、スイッチ 318 の出力は、コントローラ 300 によって使用するために新しいプロセス変数測定値を利用することができるコントローラ繰り返し中にだけ、フィルタ 316 の出力であり、別様にはゼロ (0.0) である。通信スタック 380 によって生成された新しい値フラグが設定される度に (新しいプロセス変数測定値がコントローラ 300 で利用することができることを示す)、加算器 322 は、出力をゼロに設定し、新しい期間にわたる加算を開始する。したがって、実際には、加算器 322 は、プロセス変数測定値変更の間の各コントローラ繰り返しにわたって、制御出力信号の変化を合計し、(コントローラ繰り返し中に) コントローラ 300 で新しいプロセス変数測定値更新を受信した後は常にリセットする。

#### 【0073】

フィルタ 316 によって生成されるような、制御信号の変化に対する予想されるプロセス応答は、下でより詳細に説明されるように、一次モデルを使用して近似することができる。しかしながら、より全般的には、予想されるプロセス応答は、プロセス 301 の任意の適切なモデルを使用して生成することができ、コントローラ 300 のフィードバック経路に組み込まれるモデルに、または制御信号に対する積分若しくはリセット寄与を決定することと関連付けられるフィルタまたはモデルに限定されない。例えば、予想されるプロセス応答を提供するためにモデルを利用するコントローラは、制御ルーチン 300 が PID 制御スキームを実行するように、微分寄与を組み込むことができる。例示的な微分寄与を組み込む一実施例は、図 10 に関連して下で説明される。

#### 【0074】

いずれにしても、下で説明される制御技法は、コントローラ 300 がプロセス変数の低速または非周期的更新を受信したときに、リセットまたは積分寄与を決定するためのポジティブフィードバック経路を使用することを可能にするが、それでも、新しいプロセス変数測定値の受信の間に起こる設定点変化またはフィードフォワードの変化が生じた場合に、ロバストなコントローラ応答を可能にする。具体的には、フィルタ 316 は、フィルタ 316 のこの出力が制御信号の積分成分として加算ブロック 312 に提供されたかどうかにかかわらず、コントローラ 300 のそれぞれまたは全ての実行の間、予想されるプロセス応答の新しい指示または値を算出するように構成される。その結果、コントローラ 300 が通信スタック 380 からの新しいプロセス測定値更新を受信した直後に (またはその実行サイクル中に) 生成されたフィルタ 316 の出力だけが、加算器 312 の積分またはリセット寄与として使用される場合であっても、コントローラルーチンの各実行サイクル中に、フィルタ 316 の出力が新しく再生される。

#### 【0075】

具体的には、フィルタ 316 によって生成されるような予想される応答 R の新しい指示は、現在の加算器出力 S (すなわち、最後のプロセス変数測定値更新以来の、加算器 312 によって出力された制御信号の合計した変化)、最後の (すなわち、直前の) コントローラ実行サイクル中に生成されるフィルタ 316 によって生成される予想される応答の指示、及びコントローラの実行期間から、各コントローラ実行サイクル中に算出される。その結果、各コントローラ実行サイクル中に新しいプロセス応答の推定を生成するために実行されるので、フィルタ 316 は、連続的に更新されるように本明細書で説明される。各コントローラ実行サイクル中に新しい予想されるプロセス応答またはフィルタを生成するために、連続的に更新されるフィルタ 316 によって実施することができる、例示的な式



が下に記載される。

【数 5】

$$R_N = R_{N-1} + (S_{N-1} - R_{N-1}) * \left(1 - \frac{-\Delta T}{e^{T_{Reset}}}\right)$$

式中、 $R_N$ ＝新しいフィルタ出力

$R_{N-1}$ ＝フィルタ出力の最後の実行

$S_{N-1}$ ＝積分経路加算器出力の最後の実行

$\Delta T$ ＝コントローラ実行期間

$e^{T_{Reset}}$ ＝積分期間

10

【0076】

ここで、新しいフィルタ出力  $R_N$  は、直前のフィルタ出力  $R_{N-1}$ （すなわち、現在のフィルタ出力値）に、加算器 322 からの現在のコントローラ出力値の合計した変化  $S_{N-1}$  と現在のフィルタ出力値  $R_{N-1}$  との間の差分にリセット時間  $T_{Reset}$  及びコントローラの実行期間  $T$  に依存する係数を乗じたものを加えて決定される減衰成分を加えて、繰り返し決定されることに気が付くであろう。この様式で連続的に更新するフィルタを使用することで、制御ルーチン 300 は、新しいプロセス変数測定値を受信したときに積分制御信号入力を算出したときに、予想されるプロセス応答をより良好に決定することができ、それによって、設定点の変化に対して、または 2 つのプロセス変数測定値の受信の間に起こる他のフィードフォワード外乱に対してより敏感になる。しかしながら、この積分経路の算出は、制御システムが、低速に受信した、または間欠的プロセス変数のフィードバック測定値の存在下でウィンドアップすることを防止する。より具体的には、設定点の変化（新しいプロセス測定値の受信を伴わない）が、加算器 308 の出力での誤差信号の変化を直ちにもたらし、これは、速度制御信号 326 の比例寄与成分を変化させ、したがって、制御信号 305 を変化させることに気が付くであろう。その結果、加算器 322 は、その量だけその出力  $S$  を増加させ、次いで、フィルタ 316 は、変化した制御信号に対するプロセスの新しい予想される応答を直ちに生成し始め、したがって、コントローラ 300 が新しいプロセス測定値を受信する前に出力を更新する。次いで、コントローラ 100 が新しいプロセス測定値を受信し、フィルタ出力のサンプルが制御信号の積分またはリセット寄与成分として使用されるように、スイッチ 318 によって加算器 312 の入力にクランプされたときに、フィルタ 316 は、少なくともある程度は、以前に送信した制御信号 305 に基づいて、設定点の変化に対するプロセス 301 の応答に反応した、またはそれを組み込んだ、予想されるプロセス応答を繰り返している。しかしながら、加算器 308 によって生成される誤差信号  $e$  が、プロセス変数測定値をコントローラ 300 で受信する時間の間の、プロセス変数の変化を反映することを可能にするために、この積分値は、新しい測定値を受信したときに、制御信号 326 に加えられるだけである。プロセス変数測定値を受信する時間の間のコントローラ繰り返しにおいて、加算器 312 に提供される積分要素は、ゼロに設定される。この技法は、制御システム 300 が、ウィンドアップすることを防止するか、または補助する。実際には、フィルタ 316 によって生成される積分要素は、その後のプロセス変数フィードバックを受信する時間（コントローラ繰り返し）の間のプロセス応答を推定し、そして、実際のプロセス変数応答が予想される通りであった場合に、コントローラ 300 で新しいプロセス変数測定値を受信したときに、積分成分は、比例経路において生成される値をゼロに設定する。プロセスの予想される応答が、この時間中の実際のプロセス応答とは異なる場合、積分成分は、アクチュエータを移動さ

20

30

40

50

せるように制御信号 3 2 6 を変化させ、それによって、アクチュエータの位置を修正する。

【 0 0 7 7 】

過去において、米国特許第 7 , 5 8 7 , 2 5 2 号及び第 7 , 6 2 0 , 4 6 0 号で説明されるシステムのように、非周期的に更新されるコントローラのフィードバック経路において使用されるリセット寄与フィルタは、新しいプロセス変数測定値を使用することができたときに、予想される応答の新しい指示だけを算出した。その結果、リセット寄与フィルタは、設定点変化またはフィードフォワード外乱が任意の測定値更新から完全に独立していたので、プロセス変数測定値の受信の間に起こった設定点変化またはフィードフォワード外乱を補償しなかった。例えば、設定点変化またはフィードフォワード外乱が 2 つの測定値更新の間に起こった場合は、予想される応答の新しい指示の算出が最後の測定値更新以来の時間及び現在のコントローラ出力 3 0 5 に基づいていたので、コントローラの予想されるプロセス応答が歪められる可能性があった。その結果、フィルタ 3 1 6 は、コントローラでの 2 つのプロセス変数測定値の受信の間に起こった設定点変化（または他のフィードフォワード外乱）から生じた、プロセス（または制御信号）の時間変化を補償し始めることができなかった。

10

【 0 0 7 8 】

しかしながら、理解されるように、図 9 の制御ルーチン 3 0 0 は、低速のまたは非周期的測定値に対して行われている算出に基づくことによって、予想されるプロセス応答を提供し、一方で、加えて、設定点の変化（またはコントローラ 3 0 0 に対するフィードフォワード入力として使用される任意の測定された外乱）によって引き起こされる変化を補償するために、2 つの測定値の受信の間の予想される応答を決定する。したがって、上で説明される制御技法は、予想されるプロセス応答に影響を及ぼし得る設定点変化、測定される外乱に対するフィードフォワードアクションなどに適応することができ、したがって、よりロバストな制御応答を提供する。更に、この制御技法は、コントローラにおけるウィンドアップを回避するので、プロセス変数測定値のフィードバックレートがプロセス応答時間の逆数に等しい、更にはそれよりも低いときに（すなわち、コントローラで受信しているフィードバック測定値の間の時間が、プロセス応答時間よりも長いときに）、効果的に動作することができる。

20

【 0 0 7 9 】

理解されるように、図 9 に例示される制御技法は、制御ブロックまたはルーチン 3 0 0 の各実行について連続的に更新されるフィルタ 3 1 6（例えば、リセット寄与フィルタ）を介して、予想される応答の指示を算出する。ここで、コントローラ 3 0 0 は、制御ブロックの各実行について予想される応答の新しい指示を算出するように、連続的に更新されるフィルタ 3 1 6 を構成する。しかしながら、フィルタ 3 1 6 の出力が、加算ブロック 3 1 2 への入力として使用されるべきである場合、通信スタック 3 8 0 及び、いくつかの実施例において、更新検出モジュール 8 2（図 3）は、新しいプロセス変数測定値を受信したときに、積分出力スイッチ 3 1 8 及び加算器 3 2 6 の新しい値フラグを生成するために、送信機 3 0 6 からの入来データを処理する。この新しい値フラグは、加算器 3 1 2 の入力に対するこのコントローラ繰り返しについてフィルタ出力値をサンプリングし、クランプすることをスイッチ 3 1 8 に通知する。別様には、スイッチ 3 1 8 は、積分寄与値としてゼロ（0 . 0）値を加算器 3 1 2 に提供する。

30

40

【 0 0 8 0 】

新しい値フラグが通信されるかどうかにかかわらず、連続的に更新されるフィルタ 3 1 6 は、制御ルーチンの各繰り返しについて、予想される応答の指示を算出し続ける。予想される応答のこの新しい指示は、制御ブロックの各実行について、積分出力スイッチ 3 1 8 に搬送される。新しい値フラグの存在に応じて、積分出力スイッチ 3 1 8 は、連続的に更新されるフィルタ 3 1 6 からの予想される応答の新しい指示を加算ブロック 3 1 2 に通過させることを可能にすることと、または加算ブロック 3 1 2 への入力時にゼロ値を維持することとの間で切り替える。より具体的には、新しい値フラグが通信されるときに、積分出力スイッチ 3 1 8 は、連続的に更新されるフィルタ 3 1 6 からの予想される応答の直

50

前にまたは現在算出された指示を、加算ブロック 3 1 2 に渡すことを可能にする。その逆に、新しい値フラグが存在しない場合、積分出力スイッチ 3 1 8 は、ゼロ値を加算器 3 1 2 に提供する。

#### 【 0 0 8 1 】

コントローラ 3 0 0 で新しいプロセス変数測定値を受信し、加算器 3 1 2 で連続フィルタ 3 1 6 の出力 R を使用した後または使用するとき、最後の通信以来の時間は、ゼロ ( 0 ) に設定され、連続的なフィルタ出力 R は、ゼロに設定される。同様に、加算器 3 2 2 の出力は、ゼロ ( 0 ) に設定される。更に、これらの状況において、加算器 3 1 2 は、ブロック 3 2 0 が差分の算出を行う様式または順序に応じて、新しい制御信号 3 2 6 を発生させるために、ブロック 3 2 0 の出力から連続的なフィルタ出力 R を減算することができる。

10

#### 【 0 0 8 2 】

この制御技法は、新しい測定値が通信されるかどうかにかかわらず、連続的に更新されるフィルタ 3 1 6 が、予想されるプロセス応答をモデル化し続けることを可能にする。制御出力が設定点変化または測定された外乱に基づくフィードフォワードアクションの結果として変化する場合、新しい値フラグの存在にかかわらず、連続的に更新されるフィルタ 3 1 6 は、各制御ルーチン繰り返しにおいて、予想される応答の新しい指示を算出することによって、予想されるプロセス応答を反映する。しかしながら、予想される応答の新しい指示 ( すなわち、リセット寄与または積分成分 ) は、新しい値フラグが ( 積分出力スイッチ 3 1 8 を介して ) 通信されたときに、コントローラ出力信号の算出にだけ組み込まれ、これは、コントローラ 3 0 0 で低速に受信されたプロセス変数測定値にตอบสนองして、コントローラがウィンドアップするのを防止するか、または減少させる。

20

#### 【 0 0 8 3 】

図 9 の簡単な P I コントローラ構成は、フィルタ 3 1 6 の出力を制御信号に対するリセット寄与として直接使用し、この事例において、閉ループ制御ルーチンのリセット寄与 ( 例えば、上で提示される、連続的に更新されるフィルタの式 ) は、プロセスが定常状態動作を呈するかどうかを決定するプロセス応答の正確な表現を提供することができることに留意されたい。しかしながら、むだ時間が支配的なプロセスなどの他のプロセスは、予想されるプロセス応答をモデル化するために、むだ時間ユニットを積分算出経路に含むことによって、図 5 及び図 6 に例示されるように、追加的な構成要素を図 9 のコントローラに組み込むことを必要とする場合がある。一次モデルによって十分に表されるプロセスに関しては、全般的に、P I ( または P I D ) コントローラのリセット時間を決定するために、プロセス時定数を使用することができる。より具体的には、リセット時間がプロセス時定数に等しく設定された場合、リセット寄与は、全般的に、経時的に制御ルーチン 3 0 0 が予想されるプロセス応答を反映するように、比例寄与を無効にする。図 9 に例示される実施例において、リセット寄与は、プロセス時定数と同じ時定数を有するフィルタを有するポジティブフィードバックネットワークによって達成することができる。他のモデルを使用することができるが、ポジティブフィードバックネットワーク、フィルタ、またはモデルは、既知または近似のプロセス時定数を有するプロセスの予想される応答を決定するための好都合な機構を提供する。P I D 制御を必要とするプロセスの場合、P I D 出力に対する、レートとしても知られる微分寄与は、新しい測定値を受信したときにだけ再計算し、更新することもできる。そのような事例において、微分算出は、最後の新しい測定以来の経過時間を使用することができる。プロセス測定値の非周期的受信を使用してより複雑なプロセスを制御するために他のコントローラ成分を使用することができるが、設定点変化にตอบสนองしてロバストな制御を提供するために図 9 のフィルタリング技法を使用することができる、コントローラの一実施例が、図 1 0 と併せて下で説明される。

30

40

#### 【 0 0 8 4 】

具体的には、図 1 0 は、代替のコントローラ ( または制御要素 ) 4 0 0 を描写するが、微分またはレート寄与成分がコントローラ 4 0 0 に組み込まれる点で、図 9 において上で説明されるコントローラ 3 0 0 と異なる。微分寄与を組み込むことによって、コントロー

50

ラ 4 0 0 によって実施される制御ルーチンは、いくつかの事例において、比例 - 積分 - 微分 ( P I D ) 制御スキームが実施されるように、追加的なフィードバック機構を含む。

#### 【 0 0 8 5 】

図 1 0 の制御ルーチンまたは技法は、図 7 及び図 8 のシステムと関連して上で説明される様式に類似する様式で構成される微分寄与を含み、プロセス変数測定値の低速の、非周期的な、または別様には利用できない更新に適応する。微分寄与は、最後の測定値更新以来の経過時間に基づいて再構成することができる。この様式で、微分寄与 ( 及び結果として生じる出力信号 ) のスパイクが回避される。より具体的には、図 1 0 のシステムにおける微分寄与は、微分ブロック 4 3 2 によって決定され、該微分ブロックは、比例及び積分寄与専用の要素と並列にゲインブロック 3 1 0 から ( 比例ゲイン  $K_p$  を乗じた ) 誤差信号を受信し、微分制御成分  $O_D$  を生成するように動作し、該微分制御成分は、次いで変化ブロック 4 3 3 に提供される。変化ブロック 4 3 3 は、最後のコントローラ繰り返し以来の微分制御成分  $O_D$  の変化を決定し、この変化を加算器 4 3 4 に提供し、該加算器は、この微分制御成分の変化を加算器 3 1 2 の出力に合計するか、または加えて、制御信号の変化を生成する。この事例において、積分寄与算出経路の加算器 3 2 2 は、加算器 4 3 4 の出力に接続される。しかしながら、図 9 にも例示される図 1 0 の構成要素は、図 9 に関して説明されるような様式で動作する。ここで、微分ブロック 4 3 2 は、( コントローラでプロセス変数測定値の新しい値を受信したことを示す ) 新しい値フラグをそこで受信するコントローラ繰り返し中にだけ、新しい微分成分  $O_D$  を算出するようにだけ動作することが分かるであろう。この動作は、実際には、コントローラ 4 0 0 で新しいプロセス変数測定値を受信しない間の、またはそのときの全てのコントローラ繰り返しに対して、変化ブロック 4 3 4 の出力をゼロに保つ。

#### 【 0 0 8 6 】

信頼できない伝送及び、より全般的には、測定値更新の利用不能に適応するために、微分寄与  $O_D$  は、通信スタック 3 8 0 からの新しい値フラグによって示されるように、測定値更新を受信するまで、最後に決定された値に維持される。この技法は、制御ルーチンが、制御ルーチンの通常の、または確立された実行レートに従って周期的実行を続けることを可能にする。更新された測定値を受信すると、図 1 0 に例示されるように、微分ブロック 4 3 2 は、以下の式に従って微分寄与を決定することができる。

#### 【 数 6 】

$$O_D = K_p K_D \cdot \frac{e_N - e_{N-1}}{\Delta T}$$

式中、 $e_N$  - 現在の誤差

$e_{N-1}$  = 最後の誤差

$\Delta T$  = 新しい値を通信して以来の経過時間

$O_D$  = コントローラの微分項

$K_p$  = 比例ゲイン係数

$K_D$  = 微分ゲイン係数

#### 【 0 0 8 7 】

当然、所望であれば、微分成分算出ブロック 4 3 2 は、誤差信号を受信するために、加算器 3 0 8 の出力に直接接続することができ、微分ゲイン項  $K_D$  は、比例ゲイン  $K_p$  を有する微分ゲインを組み込むように設定することができる。微分寄与を決定するためのこの技法によって、プロセス変数 ( すなわち、制御入力 ) に対する測定値更新を、出力スパイクを生成することなく、1 つ以上の実行連続期間にわたって喪失する、または利用できな

いようにすることができ、これは、バンプレスな復旧を可能にする。通信が再確立されたときに、またはコントローラで新しいプロセス変数測定値を受信したときに、微分寄与式の項 ( $e_N - e_{N-1}$ ) は、微分寄与の標準的な算出において発生させた値と同じ値を発生させることができる。しかしながら、標準的なPID技法の場合、微分寄与を決定する際の除数は、実行期間である。対照的に、本制御技法は、2つの成功裏に受信した測定値の間の経過時間を利用する。実行期間よりも長い経過時間を使用することで、制御技法は、標準PID技法よりも小さい微分寄与を生成し、また、スパイクが低減される。

#### 【0088】

経過時間の決定を容易にするために、通信スタック80は、図10に示されるように、上で説明される新しい値フラグを微分ブロック432に提供することができる。代替の実施例は、その値に基づいて、新しい測定値または更新の検出を含むこと、または関与させることができる。また、プロセス測定値を、比例または微分成分の算出において誤差の代わりに使用することができる。より全般的には、通信スタック380は、プロセス301内の任意のフィールドデバイス、コントローラの外部のプロセス制御要素などを含む、プロセス301との通信インターフェースを実施するために、任意のソフトウェア、ハードウェア、またはファームウェア（またはそれらの任意の組み合わせ）を含むこと、または関与させることができる。しかしながら、図10のコントローラ400において、連続的に更新されるフィルタ316及びスイッチモジュール318は、設定点変化に応答してロバストな制御を提供するために、図9のコントローラ300に関して上で説明されるものと同じように動作する。

#### 【0089】

本明細書で説明される制御方法、システム、及び技法の実践は、任意の1つの特定の無線アーキテクチャまたは通信プロトコルに限定されない。適切な例示的なアーキテクチャ及び通信サポートスキームは、2005年6月17日に出願された米国特許出願第11/156,215号、名称「Wireless Architecture and Support for Process Control Systems」で説明されており、その開示全体が参照により本明細書に組み込まれる。実際に、制御ルーチンに対する修正は、制御ルーチンが周期的な方法で実施されるが、各制御繰り返しに対するプロセス変数測定値を伴わない、任意の状況に十分適している。他の例示的な状況としては、サンプリングされた値が、例えばアナライザによって、または実験室サンプルを介して、不規則に提供されるか、または減多に提供されない場合が挙げられる。

#### 【0090】

本制御技法の履行は、単一入力、単一出力のPID制御ルーチン（PI及びPDルーチンを含む）とともに使用することに限定されず、むしろ、多数の異なる多重入力及び/または多重出力制御スキーム、並びにカスケード型制御スキームに適用することができる。より全般的には、本制御技法はまた、1つ以上のプロセス変数、1つ以上のプロセス入力、またはモデル予測制御（MPC）などの他の制御信号に関与する、任意の閉ループモデルに基づく制御ルーチンの状況に適用することもできる。

#### 【0091】

図11～図14は、本明細書で説明される制御ルーチンのシミュレーションされた動作の（具体的には、図10の動作の）グラフィカルな描写を提供するものであり、プロセス応答時間が、制御されたプロセス変数測定値の更新の間の時間に類似するか、更にはそれよりも短い状況における本制御ルーチンの有効性を例示するために、標準的な速度の形態のPID制御アルゴリズムを使用した従来技術のコントローラと比較している。図11～図14のグラフは、オーバーライド制御などの他のタイプの制御を使用することができるが、一次制御だけを使用した、シミュレーションされた制御の実施例を例示する。全般的に言えば、図11A、図12A、図13A、及び図14Aのグラフの各々は、標準的な従来技術の速度に基づくPID制御アルゴリズムの動作を例示するものであり、該アルゴリズムは、新しいプロセス変数測定値を各コントローラ繰り返し中に利用することができる有線フィードバック構成（その動作は、グラフの左側に例示される）、及び無線構成（そ

10

20

30

40

50

の動作は、グラフの右側に例示される)の双方を使用する。しかしながら、図11A及び図13Aは、プロセス変数測定値の間のフィードバックレートまたは時間が8秒で、プロセス応答時間が8秒である制御状況を例示し、一方で、図12A及び図14Aは、プロセス変数測定値の間のフィードバックレートまたは時間が8秒で、プロセス応答時間が3秒である、従来技術のコントローラの動作を例示する。同様に、図11及び図12は、設定点変化に应答するコントローラ動作を例示し、一方で、図13及び図14は、外乱変化に应答するこれらの同じコントローラの動作を例示する。比較するために、図11B、図12B、図13B、及び図14Bのグラフは、それぞれ、図11A、図12A、図13A、及び図14Aと同じプロセス制御状況における、本明細書で説明される速度に基づくPIDアルゴリズムの動作を例示する。

10

#### 【0092】

全般的に言えば、以下のパラメータは、図11～図14に描写されるシミュレーションされた制御動作において使用したものであり、これらの試験は、上で説明されるように、有線及び無線入力に対して、並びに設定点の変化及び測定されていない外乱に対して行った。試験で使用する制御及びプロセスのシミュレーションは、以下のようにセットアップした。

#### 8秒のプロセス応答に対する試験

一次プロセス(同じゲイン及び動力学)

プロセスゲイン = 1

プロセス時間定数 = 8秒

プロセスむだ時間 = 0秒

一次に対するPIDの調整(ラムダ係数1.0)

比例ゲイン = 1

積分ゲイン = 7.5 繰り返し/分

20

#### 3秒のプロセス応答に対する試験

一次プロセス(同じゲイン及び動力学)

プロセスゲイン = 1

プロセス時間定数 = 3秒

プロセスむだ時間 = 0秒

一次に対するPIDの調整(ラムダ係数1.0)

比例ゲイン = 1

積分ゲイン = 2.0 繰り返し/分

30

#### モジュール実行レート

全ての試験について0.5秒

#### 無線通信更新レート

全ての試験について8秒周期

#### 外乱入力

一次測定値にだけ影響

ゲイン = 1

#### 【0093】

図11A及び図13Aに例示されるように、従来技術の速度に基づくPID制御アルゴリズムは、プロセス応答時間がプロセス変数測定値の間隔に等しい(どちらも8秒に設定される)有線構成及び無線構成の双方において、設定点変化(図11A)に应答して、及び外乱変化(図13A)に対して、多少は満足に動作している。しかしながら、図11A及び図13Aのグラフの円で囲んだ部分によって示されるように、この制御技法は、無線制御中の応答中に、弁位置に大幅な変動を引き起こす。図11B及び図13Bに例示されるように、本明細書で説明される制御技法は、これらの状況(図11Bの設定点変化及び図13Bの外乱変化)においていくらか良好に動作し、有線構成の動作に非常に類似している。

40

#### 【0094】

50

しかしながら、図 1 2 A 及び図 1 4 A に例示されるように、従来技術の速度に基づく P I D コントローラは、プロセス応答時間が 3 秒で、プロセス変数更新レートが 8 秒であるときに、設定点変化及び外乱変化の双方にตอบสนองした無線制御中に、あまり十分に動作せず、実際に、不安定になる。しかしながら、図 1 2 B 及び図 1 4 B に例示されるように、現在の速度に基づく制御ルーチンは、それでも、これらの状況において非常に満足に動作し、プロセス変数測定の間隔時間が、プロセス応答時間よりも大きく（長く）、更には大幅に大きい（例えば、2 ～ 4 倍）ときに、現在説明されている制御ルーチンの有効性を例示している。

【 0 0 9 5 】

「フィールドデバイス」という用語は、本明細書において、いくつかのデバイスまたはデバイスの組み合わせ（すなわち、送信機 / アクチュエータのハイブリッドなどの、複数の機能を提供するデバイス）、並びに制御システムにおいて機能を行う任意の他のデバイス（複数可）を含むように、広義に使用される。いずれにしても、フィールドデバイスは、例えば、入力デバイス（例えば、温度、圧力、流量などのプロセス制御パラメータを示すステータス、測定値、または他の信号を提供するセンサ及び機器などのデバイス）、並びにコントローラ及び / または弁、スイッチ、流れ制御デバイスなどの他のフィールドデバイスから受信したコマンドにตอบสนองしてアクションを行う制御オペレータまたはアクチュエータを含むことができる。

【 0 0 9 6 】

本明細書で説明される任意の制御ルーチンまたはモジュールは、複数のデバイスにわたって分散した状態で実施または実行する、その一部を有することができることに留意されたい。その結果、制御ルーチンまたはモジュールは、所望に応じて、異なるコントローラ、フィールドデバイス（例えば、スマートフィールドデバイス）若しくは他のデバイス、または他の制御要素によって実施される一部分を有することができる。同様に、本明細書で説明される、プロセス制御システム内で実施される制御ルーチンまたはモジュールは、ソフトウェア、ファームウェア、ハードウェア等を含む任意の形式をとることができる。そのような機能性を提供することに関与する任意のデバイスまたは要素は、ソフトウェア、ファームウェア、またはそれに関連付けられるハードウェアが、プロセス制御システム内のコントローラ、フィールドデバイスに配置されるか、または任意の他のデバイス（またはデバイスの集合体）に配置されるかにかかわらず、本明細書では、一般に、「制御要素」と称されることがある。制御モジュールは、例えば、任意のコンピュータ読み出し可能な媒体に記憶されるルーチン、ブロック、またはその任意の要素を含む、プロセス制御システムの任意の一部または一部分とすることができる。そのような制御モジュール、制御ルーチン、またはその任意の一部（例えば、ブロック）は、本明細書で一般的に制御要素と称される、プロセス制御システムの任意の要素またはデバイスによって実施または実行することができる。モジュールとするか、またはサブルーチン、サブルーチンの一部（一連のコードなど）などの制御手順の任意の一部とすることができる制御ルーチンは、オブジェクト指向プログラミングを使用して、またはラダーロジック、シーケンシャル機能チャート、機能ブロック図を使用して、または任意の他のソフトウェアプログラム言語若しくは設計パラダイムを使用して、任意の所望のソフトウェアフォーマットで実施することができる。同様に、制御ルーチンは、例えば、1 つ以上の E P R O M、E E P R O M、特定用途向け集積回路（A S I C）、または任意の他のハードウェア要素若しくはファームウェア要素にハードコードすることができる。なお更に、制御ルーチンは、グラフィカル設計ツール、または任意の他のタイプのソフトウェア / ハードウェア / ファームウェアプログラミング若しくは設計ツールを含む任意の設計ツールを使用して設計することができる。したがって、コントローラ 1 1 は、制御ストラテジまたは制御ルーチンを任意の所望の様式で実施するように構成することができる。

【 0 0 9 7 】

代替的にまたは追加的に、機能ブロックは、フィールドデバイス自体またはプロセス制御システムの他の制御要素に記憶し、それらによって実施することができるが、それは、

10

20

30

40

50

システムがフィールドバスデバイスを利用する場合であり得る。制御システムの説明は、機能ブロック制御ストラテジを使用して本明細書で提供されるが、制御技法及びシステムはまた、ラダーロジック、シーケンシャル機能チャートなどの他の規則を使用しても、または任意の他の所望のプログラム言語またはパラダイムを使用しても、実施または設計することができる。

【 0 0 9 8 】

実施するときに、本明細書で説明されるソフトウェアのいずれかは、磁気ディスク、レーザーディスク（登録商標）、または他の記憶媒体、コンピュータ若しくはプロセッサのRAM若しくはROMなどの、任意のコンピュータ読み出し可能なメモリに記憶することができる。同様に、このソフトウェアは、例えば、コンピュータ読み出し可能なディスク若しくは他の可搬型コンピュータ記憶機構、または電話回線、インターネット、ワールドワイドウェブ、任意の他のローカルエリアネットワーク、若しくはワイドエリアネットワークなどの通信チャネルを含む、任意の既知の、または所望の配信方法を使用して、ユーザ、プロセスプラント、またはオペレータワークステーションに配信することができる（配信は、可搬型記憶媒体を介して、かかるソフトウェアを提供することと同じであるか、または互換性があると見なされる）。更に、このソフトウェアは、変調若しくは暗号化を伴わずに直接提供することができ、または通信チャネルを通じて送信される前に、任意の適切な変調搬送波及び／または暗号化技術を使用して、変調及び／または暗号化することができる。

【 0 0 9 9 】

このように、本発明を例示することだけを意図し、限定することは意図せずに、本発明を具体的な実施例を参照しながら説明してきたが、当業者には、本発明の趣旨及び範囲から逸脱することなく、制御技法に変更、追加、または削除を行うことができることが明らかになるであろう。

10

20

30

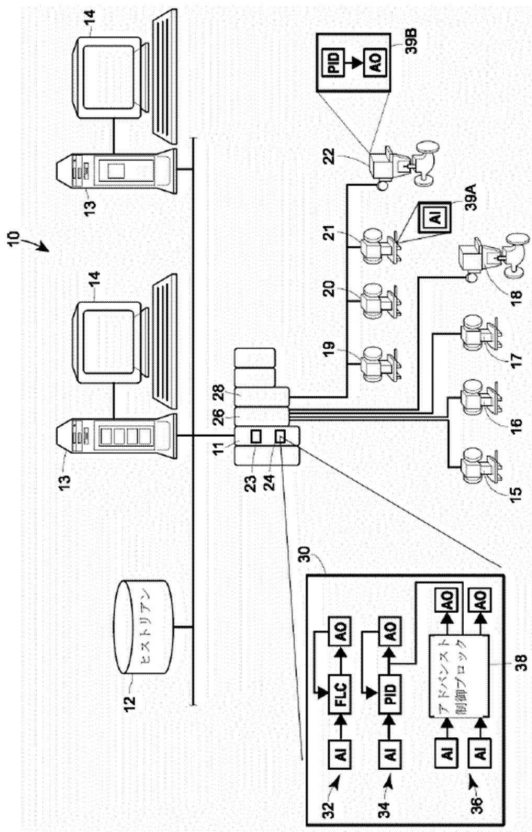
40

50

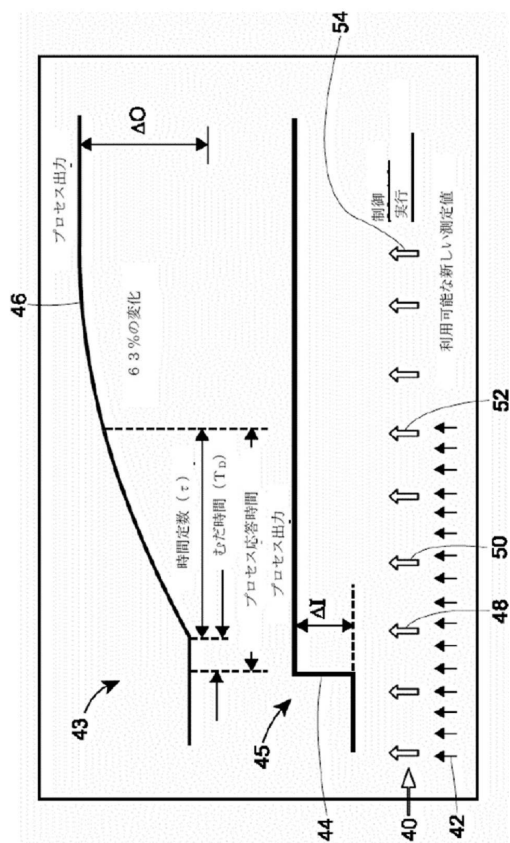


【図面】

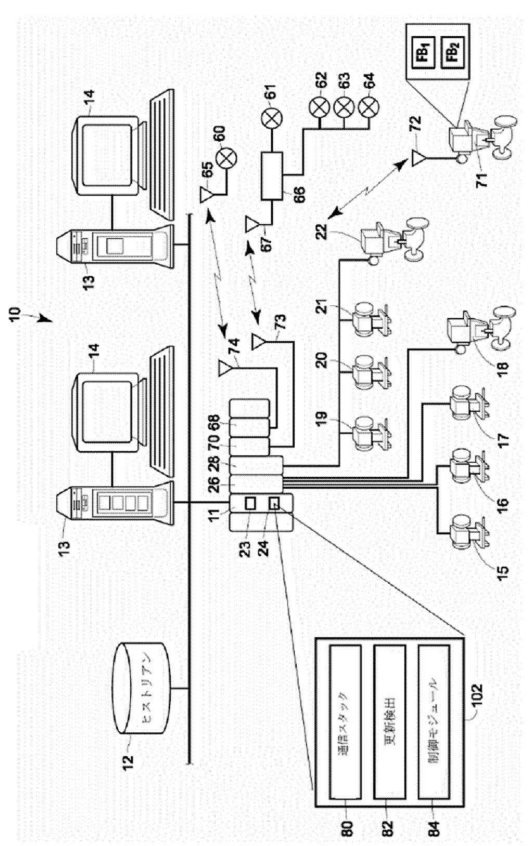
【図 1】



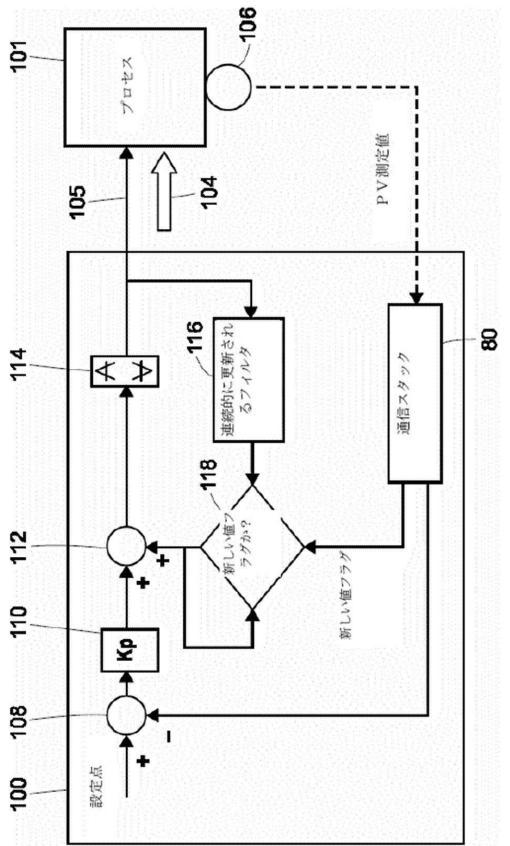
【図 2】



【図 3】



【図 4 A】



10

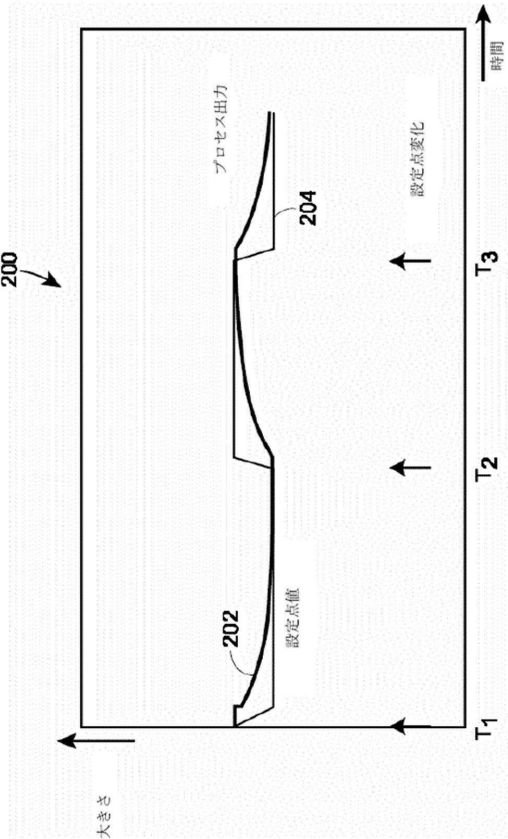
20

30

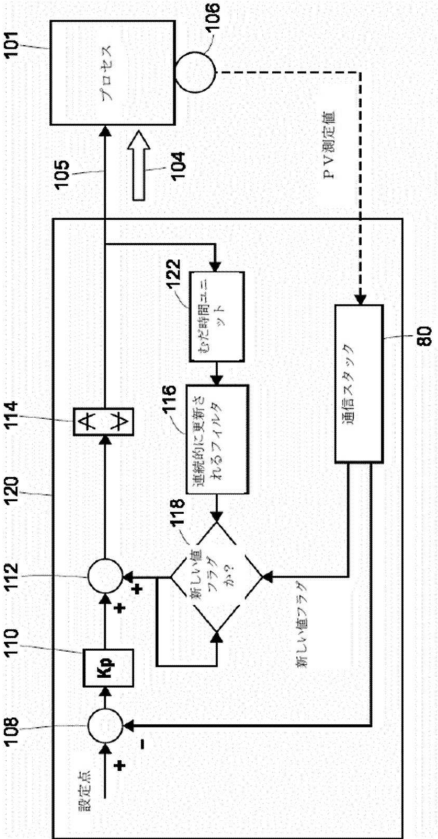
40

50

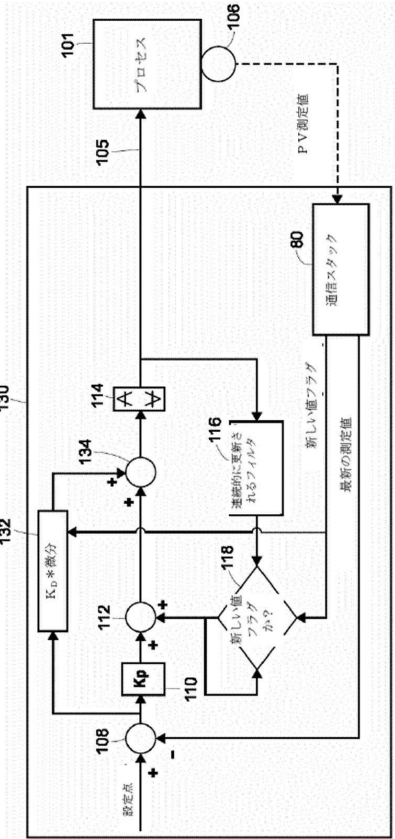
【図 4 B】



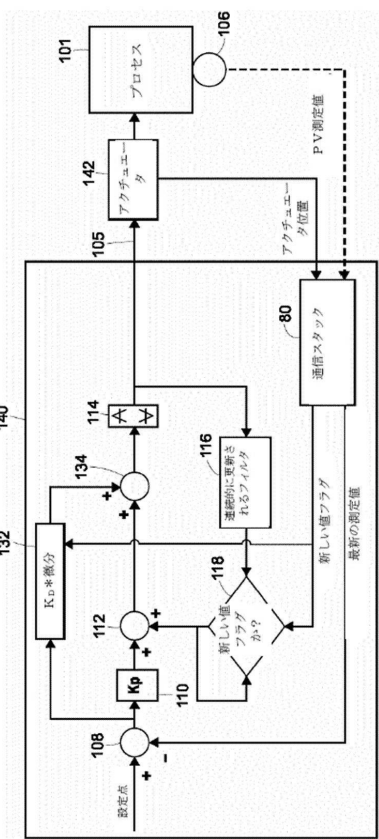
【図 5】



【図 6】



【図 7】



10

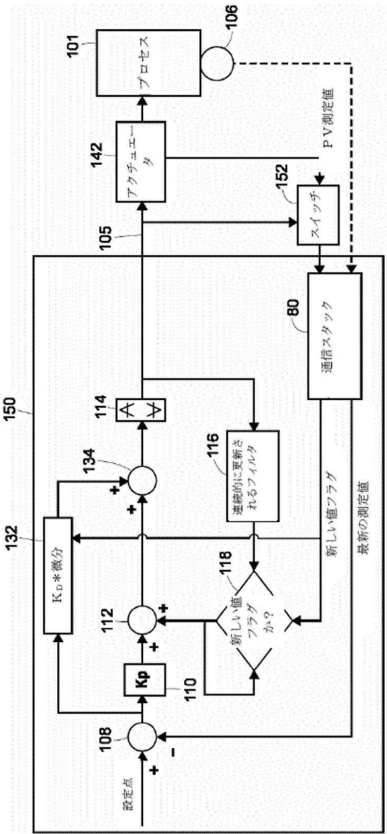
20

30

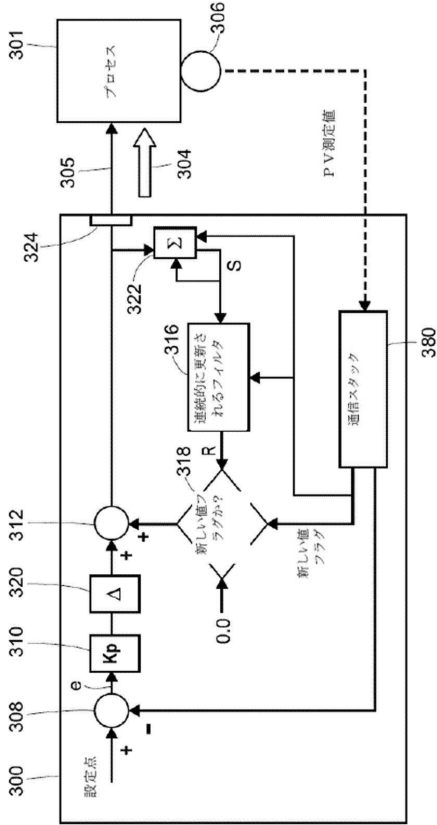
40

50

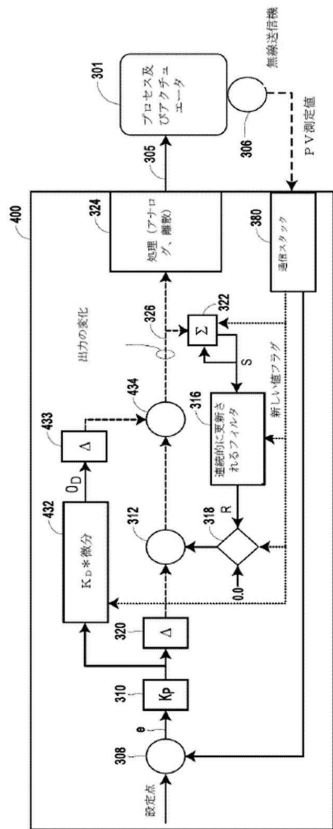
【図 8】



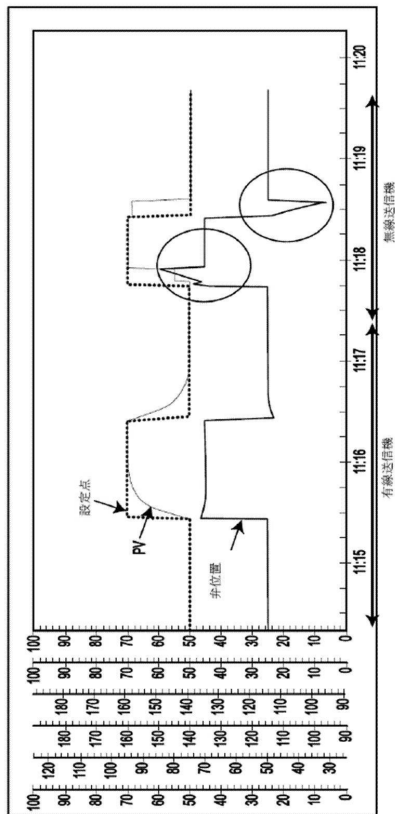
【図 9】



【図 10】



【図 11 A】



10

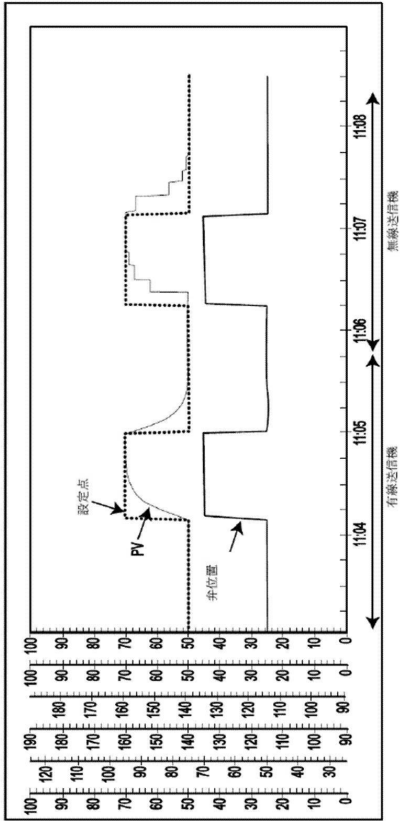
20

30

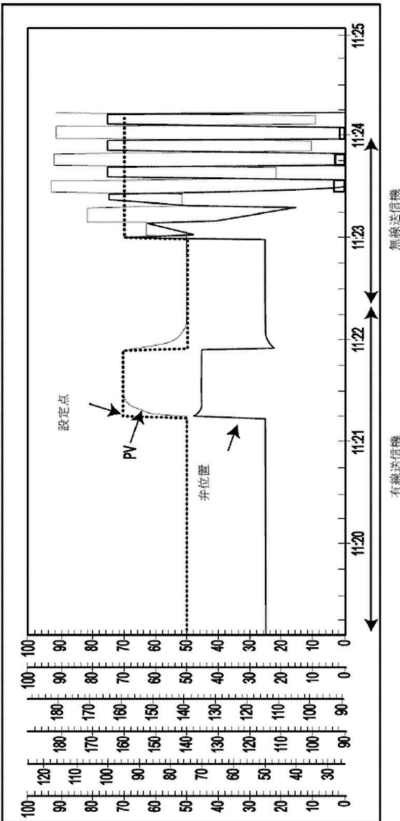
40

50

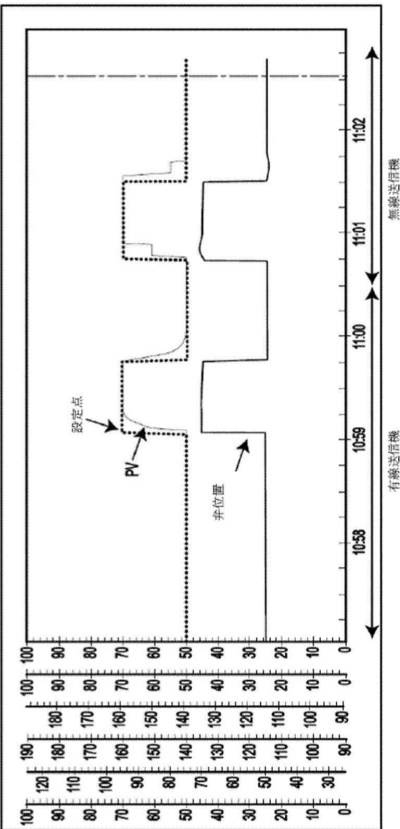
【図 1 1 B】



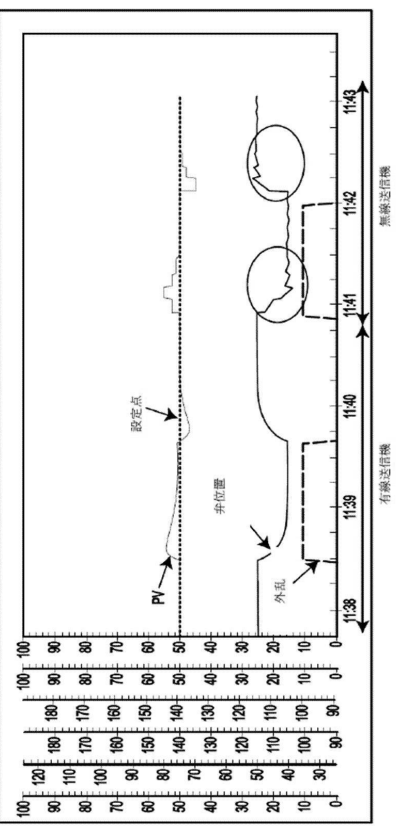
【図 1 2 A】



【図 1 2 B】



【図 1 3 A】



10

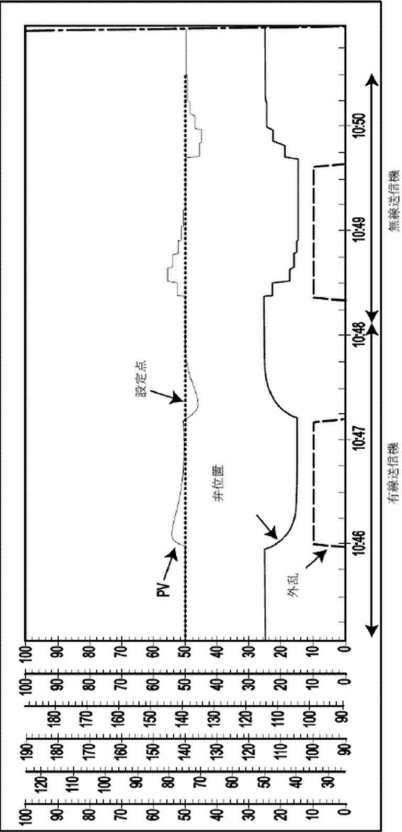
20

30

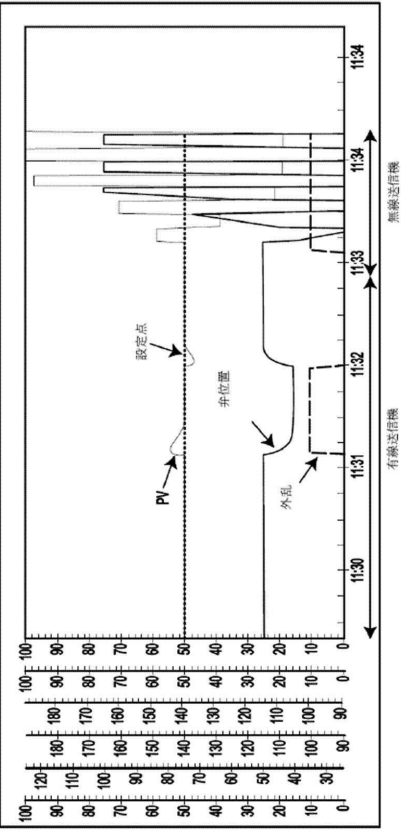
40

50

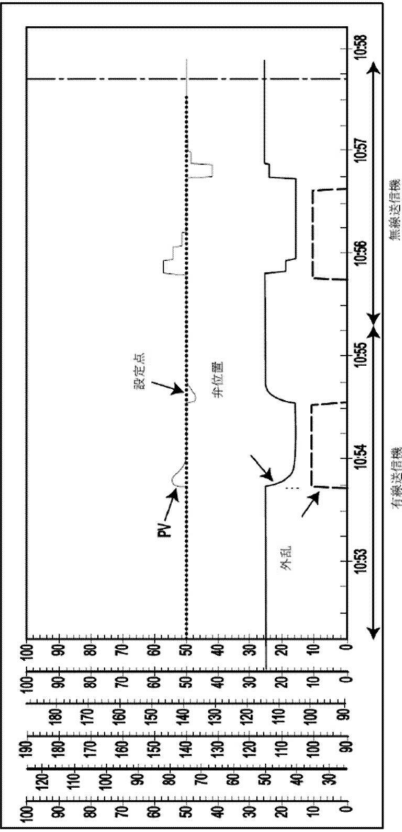
【図 1 3 B】



【図 1 4 A】



【図 1 4 B】



10

20

30

40

50

フロントページの続き

                  米国(US)  
                  アメリカ合衆国 アイオワ州 5 0 1 5 8 マーシャルタウン ローリング メドウズ ロード 1 2 1 8  
(72)発明者     マーク ジェイ .、 ニクソン  
                  アメリカ合衆国 テキサス州 7 8 6 8 1 ラウンド ロック ブラックジャック ドライブ 1 5 0 3  
合議体  
  審判長     見目 省二  
  審判官     大山 健  
  審判官     田々井 正吾  
(56)参考文献     特開 2 0 1 3 - 1 4 9 2 5 3 ( J P , A )  
                  特開昭 5 9 - 4 7 6 0 9 ( J P , A )  
                  特開平 1 0 - 3 1 5 1 6 ( J P , A )  
                  特開 2 0 1 0 - 1 7 8 5 0 9 ( J P , A )  
(58)調査した分野 (Int.Cl. , D B 名)  
                  G05B 11/42