

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4597603号
(P4597603)

(45) 発行日 平成22年12月15日(2010.12.15)

(24) 登録日 平成22年10月1日(2010.10.1)

(51) Int.Cl. F I
G 0 6 F 15/00 (2006.01) G O 6 F 15/00 3 1 0 A
G 0 5 B 23/02 (2006.01) G O 5 B 23/02 3 0 1 L

請求項の数 3 (全 13 頁)

(21) 出願番号	特願2004-226590 (P2004-226590)	(73) 特許権者	594120847
(22) 出願日	平成16年8月3日(2004.8.3)		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
(62) 分割の表示	特願2000-134873 (P2000-134873) の分割		アメリカ合衆国 78759 テキサス オースティン リサーチ パーク プラザ ビルディング 111 リサーチ ブル ーバード 12301
原出願日	平成12年5月8日(2000.5.8)	(74) 代理人	110000556
(65) 公開番号	特開2005-50358 (P2005-50358A)		特許業務法人 有古特許事務所
(43) 公開日	平成17年2月24日(2005.2.24)	(72) 発明者	マーク ジェイ. ニクソン
審査請求日	平成19年4月24日(2007.4.24)		アメリカ合衆国 78681 テキサス ラウンド ロック ブラックジャック ド ライブ 1503
(31) 優先権主張番号	60/132780		
(32) 優先日	平成11年5月6日(1999.5.6)		
(33) 優先権主張国	米国 (US)		
(31) 優先権主張番号	09/510053		
(32) 優先日	平成12年2月22日(2000.2.22)		
(33) 優先権主張国	米国 (US)		
前置審査			最終頁に続く

(54) 【発明の名称】 単一のコンピュータ上で機能的に統合された分散型処理制御システム

(57) 【特許請求の範囲】

【請求項1】

制御モジュールを用いて1個以上のフィールド装置を制御し第1の通信プロトコルに従って通信を行う第1の分散型制御装置と、第2の通信プロトコルに従って通信を行う第2の分散型制御装置とから離れて位置するユーザワークステーションを有する分散型処理制御システムと共に使用するための装置であって、該装置が、

メモリおよび処理装置を有するコンピュータと、

前記コンピュータに接続された表示装置と、

前記コンピュータのメモリに記憶され前記コンピュータの処理装置及び前記第1の制御装置で実行されるように構成された制御装置アプリケーションとを備え、前記制御装置アプリケーションは、前記第1の分散型制御装置によって実行されるように構成された1つ以上の制御モジュール又は前記分散型処理制御システムの動作中に前記第1の分散型制御装置によって制御される1つ以上のフィールドデバイスとを包含し、前記制御装置アプリケーションは、前記制御装置アプリケーションが前記コンピュータの処理装置で実行されたときに実行される1つ以上の陰機能ブロックをさらに包含し、該陰機能ブロックは、前記第1の通信プロトコルと前記第2の通信プロトコルとの間で、制御パラメータ、値又はコマンドをマッピングし、前記制御装置アプリケーションがデータを前記第2の分散型制御装置に供給し及びこれからコマンド及びデータを受け取るようにするように構成され、

前記コンピュータのメモリに記憶され、前記コンピュータの処理装置で実行されたときに前記制御装置アプリケーションと通信し、前記表示装置を用いて前記第2の分散型制御

装置から送られた情報を表示するように前記表示装置を使用する表示アプリケーションとを備えていることを特徴とする、装置。

【請求項 2】

請求項 1 に記載の装置において、前記第 2 の分散型制御装置と前記コンピュータとの間に接続されたインタフェースをさらに備えることを特徴とする、装置。

【請求項 3】

請求項 2 に記載の装置において、前記インタフェースが O P C インタフェースであることを特徴とする、装置。

【発明の詳細な説明】

【技術分野】

10

【0001】

本発明は、一般に処理制御システムに関し、特に単一コンピュータ上の分散型処理制御システムに関連する処理制御機能性を統合するシステムに関する。

【背景技術】

【0002】

化学、石油または他の処理で使用する分散型処理制御システムは、一般に、アナログ、デジタルまたは複合アナログ/デジタルバスを介して、1個以上のフィールド装置に通信結合された1個以上の処理制御装置を備える。弁、弁位置決め装置、スイッチ、送信器（例えば、温度、圧力および流量センサ）等のフィールド装置が、処理環境内に配置され、開閉弁、測定処理パラメータ等の処理機能を実施する。周知のフィールドバスプロトコルに従うフィールド装置等の高性能フィールド装置はまた、制御計算、警報機能、一般に制御装置内で実現される他の制御機能を実施することも可能である。一般にプラント環境内でも配置される処理制御装置は、フィールド装置によって行われる処理測定を示す信号およびフィールド装置に関する他の情報またはそのいずれかを受信し、例えば、処理制御判断を行い、受信した情報に基づいて制御信号を生成し、フィールドバスフィールド装置等のフィールド装置において実施される制御モジュールまたはブロックと統合する異なった制御モジュールを実行する制御装置アプリケーションを実施する。制御装置内の制御モジュールは、フィールド装置への通信回線上に制御信号を送信して処理動作を制御する。

20

【0003】

30

フィールド装置および制御装置からの情報は、一般に配電盤室またはハッシュプラント環境から離れた他の場所に配置されたオペレータワークステーション、パーソナルコンピュータ、データヒストリアン、報告書作成プログラム、集中データベース等、1個以上の他のハードウェア装置に至るデータハイウェイ上で通常入手可能である。これらのハードウェア装置は、例えば、処理制御ルーチンの設定変更、制御装置またはフィールド装置内の制御モジュールの動作の変更、現在の処理状態の表示、訓練者用処理動作のシミュレートまたは処理制御ソフトウェアのテスト、構成データベースの維持および更新等、処理に関する機能をオペレータが実施可能なアプリケーションを実行する。

【0004】

例えば、フィッシャ・ローズマウント・システムズ・インク（F i s h e r - R o s e m o u n t S y s t e m s , I n c . ）販売のデルタ V T M 制御システムは、処理内の様々な場所に配置された異なる装置内に記憶され且つそれによって実行される複数のアプリケーションを備える。1個以上のオペレータワークステーション内の構成アプリケーションにより、ユーザは、処理制御モジュールを作成または変更し、専用分散型制御装置に至るデータハイウェイを介してこれらの処理制御モジュールをダウンロードすることが可能である。構成アプリケーションにより、設計者は、表示アプリケーションによって使用するユーザインタフェースを作成または変更し、データをユーザに表示し、それによって、処理制御ルーチン内での設定点等の設定をユーザが変更できるようにする。専用制御装置やフィールド装置もまた、割り当てられてダウンロードされた制御モジュールを実行して実際の処理制御を実現する制御装置アプリケーションを記憶し、実行する。1個以上の

40

50

オペレータワークステーションで実行可能な表示アプリケーションは、データハイウェイを介して制御装置アプリケーションからデータを受信し、このデータを、処理制御システム的设计者、オペレータまたはユーザインタフェースを使用するユーザに表示し、それによって、オペレータの表示、エンジニアの表示、技術者の表示等の多数の異なる表示のうちいずれかを提供し得る。データヒストリアンアプリケーションは、一般に、データハイウェイを横切るデータのうちのいくつか、またはその全てを収集して記憶するデータヒストリアン装置に記憶され、実行され、構成データベースアプリケーションは、データハイウェイに装着され、現在の処理制御ルーチン構成およびそれに関連するデータを記憶するさらなるコンピュータ装置において実行可能である。あるいは、構成データベースは、構成アプリケーションとして同一ワークステーションに記憶される。

10

【0005】

ミズーリー州、セントルイスのダン・エイチ・マンガー・カンパニー (Don H. Munger Company) 販売のミミックアプリケーションやカナダ、カルガリーのハイプロテック (Hyprotech) 製造販売のHYSYSアプリケーション等のシミュレーションアプリケーションは、データハイウェイに装着されたパーソナルコンピュータ上で実行可能である。これらの、または他のインタフェースアプリケーションは、データハイウェイを介して制御装置アプリケーションおよび表示アプリケーションと通信し、制御装置に記憶された制御モジュールまたはワークステーションで実行する表示アプリケーションによって使用するユーザインタフェースのテストを可能にする。これらのシミュレーションアプリケーションはまた、訓練およびシステムチェックアウトを実施可能にする。高忠実度処理シミュレーション製品であるHYSYSアプリケーションは、ミミックアプリケーションとして同一コンピュータに統合され、処理制御設計をチェックアウトするためのその正確なタイミングおよび応答をシミュレートする場合もあった。一般に、これらのシミュレーションアプリケーションは、周知のOPCインタフェースまたはPIインタフェース等の標準インタフェースを用いて、制御装置またはフィールド装置内で制御装置アプリケーションと通信する。あるいは、シミュレーションは手動で行われ、オペレータまたは他のユーザは、制御装置またはフィールド装置の制御モジュール内で入力/出力 (I/O) 機能ブロックのシミュレート能力を利用して、I/Oブロックによって処理される値や状態変数を手動で設定し、フィールド装置を実際に制御することなく、またはフィールド装置を制御装置に接続することなく、表示情報論理または制御論理を検証可能にする。いずれかの場合、特定のシミュレーションモジュールを作成して制御装置またはフィールド装置に記憶し、出力ブロックターゲットに基づくシミュレートされた処理ダイナミクスを実制御モジュールに与えることによって実制御モジュールと相互作用することも可能である。

20

30

【0006】

これらのシステムでは、表示アプリケーションによって生成されるユーザ表示に示される情報が、制御モジュールを設計して、それを制御装置またはフィールド装置に割り当てダウンロードし、シミュレーション手順を使用してシステムをテストした後、制御装置によって正しく通信されているかどうかを検証することができる。構成チェックアウトおよびオペレータ訓練のための完全システムハードウェアを使用することによって、制御装置への物理的接続および制御装置とワークステーションとの通信が十分にテスト可能であるという利点がある。また、性能、所要メモリ量、および構成モジュールにより制御装置内に導入されたロードを調べることも可能である。しかし、いずれの場合でも制御論理または表示論理のチェックアウトを可能にするために、ハードウェア制御装置および高性能フィールド装置またはそのいずれかが存在、すなわちシステムに接続されなければならない。その理由は、制御装置アプリケーションおよび処理制御モジュールが、制御装置または制御装置に接続されたフィールド装置内で実行するために作成され、記憶されるためである。

40

【0007】

しかし、多くの場合、プラントおよび制御設計が完全になるまで分散型処理制御システ

50

ム用機器を購入することは不可能であるため、制御装置およびフィールド装置内の制御モジュールおよびユーザワークステーション内のユーザインタフェースのテストを妨げるか、または遅延させることになる。さらに、機器を購入した後、実際に取り付ける際にタイプアップされる場合が多いため、通常、構成チェックアウトや訓練のためには利用できない。このため、多くの場合、システムハードウェアをさらに追加購入してプラント設計、構成チェックアウトおよびオペレータ訓練を支持する必要がある。この追加のハードウェアの費用は、予備の部品としての予算である場合もあるが、多くの場合、プラントは、オペレータ訓練、プラント設計および構成チェックアウトを支持するための費用として認めていない。

【 0 0 0 8 】

設計に必要なハードウェアの量を減じる処理制御システムもあり、処理制御アプリケーションが、表示アプリケーションや構成データベースアプリケーション等、他のソフトウェアとして同一ワークステーション内で実行される。しかし、これらのシステムは、正確には分散型処理制御システムではない。その理由は、処理環境から離れたオペレータまたはエンジニアステーションに処理制御ソフトウェアが配置され、一般に、より長い距離に亘って処理制御通信が行われ、処理環境内でワークステーションとI/Oおよびフィールド装置との間により長い通信回線が配設される必要があるからである。さらに、このような結合システムは、テストし、実行時に処理制御ソフトウェアを異なった専用制御装置またはフィールド装置にダウンロードするために、ワークステーション内での処理制御ソフトウェアのシミュレーションに備えていない。

【 発明の開示 】**【 発明が解決しようとする課題 】****【 0 0 0 9 】**

本発明は、分散型処理制御システムソフトウェアを容易に開発してテストし、そのソフトウェアを使用して分散型処理制御システムのシミュレーションを可能にする単一パーソナルコンピュータまたはラップトップコンピュータ等の単一コンピュータにおける分散型処理制御システム機能性の統合に関する。

【 課題を解決するための手段 】**【 0 0 1 0 】**

一実施態様では、制御モジュールを用いて1個以上のフィールド装置を制御する分散型制御装置から離れた場所にユーザワークステーションを配置した分散型処理制御システムと共に使用する装置は、メモリおよび処理装置を有するコンピュータを備える。構成アプリケーションは、コンピュータのメモリに記憶され、コンピュータの処理装置で実行される。この構成アプリケーションはまた、ユーザワークステーションで実行され、分散型制御装置によって実行するための制御モジュールを作成することが可能である。さらに、制御装置アプリケーションは、コンピュータのメモリに記憶され、コンピュータの処理装置で実行される。この制御装置アプリケーションは、さらに分散型制御装置で実行され、処理制御システムの動作中に制御モジュールのうちの1個を実現する。このシステムにおいて、構成アプリケーションは、さらにコンピュータ上で実行される時、処理制御システム内の分散型制御装置によって使用可能な第1の制御モジュールを作成し、制御装置アプリケーションは、コンピュータ内で第1の制御モジュールを実行し、それによって分散型処理制御システムの動作をシミュレートする。このシステムを使用することにより、最終的に分散型処理制御システムソフトウェアを実行する専用ハードウェア、例えば分散型処理制御装置およびフィールド装置を使用することなく、そのソフトウェアおよび制御モジュールのシミュレーションおよびテストが可能となる。

【 0 0 1 1 】

他の実施態様では、制御モジュールを用いて1個以上のフィールド装置を制御する分散型制御装置から離れた場所にユーザワークステーションが配置された分散型処理制御システムと共に使用する装置が、メモリと、処理装置と、この処理装置に接続された表示装置とを有するコンピュータを備える。制御装置アプリケーションが、コンピュータのメモリ

10

20

30

40

50

に記憶され、コンピュータの処理装置において実行される。この制御装置アプリケーションはまた、分散型制御装置において実行され、処理制御システムの動作中に制御モジュールを実現し、分散型処理制御システムの分散型制御装置とは異なる型式のさらなる制御装置と通信可能である。表示アプリケーションは、コンピュータのメモリに記憶され、コンピュータの処理装置において実行される。この表示アプリケーションはまた、コンピュータ内で制御装置アプリケーションと通信し、表示装置を用いて、さらなる制御装置から送信された情報をユーザに表示する。

【発明の効果】

【0012】

このシステムを使用することによって、最終的に分散型処理制御システムソフトウェアを実行するハードウェア、例えば分散型処理制御装置およびフィールド装置を使用することなく、そのソフトウェアおよび制御モジュールのシミュレーションおよびテストを可能にする。

10

【発明を実施するための最良の形態】

【0013】

図1を参照すると、周知の典型的分散型処理制御システム10が、1個以上の専用処理制御装置12を備えており、その各々は、例えばフィールドバスインタフェースであってもよい入力/出力(I/O)モジュール16を介して1個以上のフィールド装置14および15に接続されている。制御装置12はまた、例えばイーサネット(R)リンクであってもよいデータハイウェイ20を介して1個以上のホストまたはオペレータワークステーション18にも結合されている。制御装置12と、I/Oモジュール16と、フィールド装置14および15とは、過酷なプラント環境内に配置され、その環境を通じて分散され、配電盤室または制御装置要員によって容易にアクセス可能なあまり過酷でない他の環境にオペレータワークステーション18が通常、配置される。例えば、フィッシャ・ローズマウント・システムズ・インク(fisher-Rosemount Systems, Inc.)が販売するデルタV制御装置でもよい制御装置12の各々は、多数の異なった、独立して実行される制御モジュール24を用いて制御方針を実現する制御装置アプリケーション23を記憶し、実行する。制御モジュール24の各々は、一般に機能ブロックと呼ばれるものから構成することも可能であり、各機能ブロックは制御ルーチン全体の一部であるかまたはそのサブルーチンであり、(リンクと呼ぶ通信を介して)他の機能ブロックに関連して動作して、処理制御システム10内で処理制御ループを実現する。周知のように機能ブロックは、一般に、送信器、センサまたは他の処理パラメータ測定装置等に関連する入力機能、PID、ファジー論理等の制御を実施する制御ルーチンに関連する制御機能、または弁等の装置の動作を制御して処理制御システム10内で物理機能を実施する出力機能を実行する。当然のことながら、ハイブリッドな他の型式の機能ブロックが存在する。しかし、制御モジュール24は、例えば、逐次制御ブロック、はしご論理等を含む望ましい制御プログラミング方式を用いて設計することも可能であり、機能ブロックや他の特定のプログラミング技術を用いた設計に限定されない。

20

30

【0014】

図1に示すシステムにおいて、制御装置12のうちの1個に接続されたフィールド装置15は、プロセッサとメモリとを備えるフィールドバスフィールド装置等の高性能フィールド装置である。これらの装置は、制御装置アプリケーション23および機能ブロック等の1個以上のモジュール24の副部分を記憶して実行する。フィールド装置15内のモジュールまたはモジュールの一部は、制御装置12内でのモジュールの実行に関連して実行され、周知のように処理制御を実現することが可能である。

40

【0015】

ホストワークステーション18は、処理制御モジュール24を作成または変更し、これらの制御モジュールをデータハイウェイ20を介して制御装置12のうちの1個およびフィールド装置、例えばフィールド装置15のうちの1個またはそのいずれかにダウンロードするために使用する構成アプリケーション25を記憶し、実行することが可能である。ホ

50

ワークステーション 18 もまた、データハイウェイ 20 を介して制御装置 12 からデータを受信し、この情報を、一般に構成アプリケーション 25 を用いて作成された既定ユーザインタフェース 27 または表示を用いる表示機構を介して表示する表示アプリケーション 26 を記憶し、実行する。表示アプリケーション 26 が、設定点変更等の入力をユーザから受け取り、これらの入力を 1 個以上の制御装置 12 内で制御装置アプリケーション 23 に与える場合もある。

【 0016 】

データヒストリアン 28 がデータハイウェイ 20 に接続され、いずれかの望ましい、または既知のデータヒストリアンソフトウェアを用いてメモリにデータを記憶する。しかし、データヒストリアンは、必要に応じて 1 個以上のワークステーション 18 に設けることも可能である。さらに、構成データベース 30 は、処理制御システムの現在の構成およびそれに関連するデータを記憶する構成データベースアプリケーション 32 を実行する。

【 0017 】

パーソナルコンピュータ (PC) 34 は、データハイウェイ 20 に接続され、他のアプリケーション、例えば訓練、テストおよびシミュレーションアプリケーション 36 等を実行することも可能である。このようなアプリケーションは、一般に、処理制御システムの動作またはセットアップに関連する制御装置 12 および構成データベース 30 と相互作用し、そこからデータを受け取る。上記のように、制御装置 12 およびフィールド装置 15 内の処理制御モジュール 24 は、シミュレーションモードまたは状態に設定され、所定の、またはオペレータ提供の値をシミュレーションソフトウェア 36 に戻して、制御装置アプリケーション 23 が使用する制御モジュール 24 および表示アプリケーション 26 が使用するユーザインタフェース 27 のテストを可能にすることもできる。表示アプリケーション 26 および構成アプリケーション 25 のうちのいずれか、または両方は、PC 34 に記憶され、それによって実行され、処理制御モジュール 24 およびユーザインタフェース 27 またはそのいずれかを変更可能にする場合もある。しかし、図 1 のシステムでは、シミュレーションアプリケーション 36 を使用して、これらの構成要素をテスト可能にするまでに、制御装置アプリケーション 23 および処理制御モジュール 24 が専用制御装置 12 およびフィールド装置 15 またはそのいずれかにロードされ、実行されることが認められる。

【 0018 】

図 2 を参照すると、CPU 42 と、メモリ 44 と、ユーザ表示装置 46 とを有する 1 個のコンピュータ 40 は、図 1 の制御システムの様々な異なるハードウェアに関連する分散型処理制御システムソフトウェアを統合して記憶し、実行し、例えば、訓練、設計およびテストのための処理制御システムのシミュレーションを実施するために使用することも可能である。コンピュータ 40 は、図 1 のシステムのデータハイウェイ 20 に接続することも可能であるが、例えば、独立型装置として動作して、図 1 のシステムで使用するソフトウェア構成要素のシミュレーションを提供することも可能である。コンピュータ 40 は、いずれかの形態の表示装置、メモリ、CPU 等が対応するパーソナルコンピュータ、ラップトップコンピュータ、ワークステーション等を備えたコンピュータであってもよいが、図 1 に示す専用ハードウェア構成要素よりも一般に安価であり、入手し易い。

【 0019 】

図 2 の実施の形態では、コンピュータ 40 は、メモリ 44 における構成アプリケーション 25、表示アプリケーション 26、さらに必要に応じてシミュレーションアプリケーション 36 およびデータベース構成アプリケーション 32 のいずれか、またはその両方を記憶し、これらのアプリケーションを CPU 42 に必要なものとして実行する。コンピュータ 40 は、実行するための制御装置アプリケーション 23 も記憶していることが重要である。この場合、構成アプリケーション 25 は、(最終的に処理を制御する際に使用するための制御装置およびフィールド装置またはそのいずれかにダウンロードされる) 1 個以上の処理制御モジュール 24 を設計して作成し、さらに(最終的に表示アプリケーション 26 によってオペレータワークステーション 18 で使用する) ユーザインタフェース 27 を

10

20

30

40

50

作成するために使用可能である。しかし、処理制御モジュール 24 のうちいずれかを制御装置またはフィールド装置に割り当ててダウンロードする前に、またはユーザインタフェース 27 をワークステーションに割り当ててダウンロードする前に、これらの構成要素をテストまたはシミュレーションの目的でコンピュータ 40 に記憶するか、または割り当てる。

【0020】

制御モジュール 24 およびユーザインタフェース 27、またはそのいずれかを作成するか、または受け取った後、CPU 42 は、制御装置アプリケーション 23 および関連の制御モジュール 24 を実行し、表示アプリケーション 26 および関連のユーザインタフェース 27 を実行し、さらに、データベース構成アプリケーション 32 を実行して、例えばこれらの構成要素の動作をシミュレートすることも可能である。当然のことながら、表示アプリケーション 26 およびデータベース構成アプリケーション 32 を使用する場合には、オブジェクト指向プログラミング技術に関連するものとして、ウィンドウズ型オペレーティングシステムを使用するか、または望ましいか、もしくは標準のタイムシェアリングプロトコルを用いるようないずれか望ましい方法で、CPU 42 上で制御装置アプリケーション 23 と共に実行される。

10

【0021】

制御装置アプリケーション 23 は、I/O モジュールが付加された専用制御装置 12 ではなく、コンピュータ 40 内で実行中であるため、制御モジュール 24 の入力および出力ブロックは、フィールド装置または入力/出力モジュール等の付加装置の動作をシミュレートするために設定されなければならない。これは、図 1 のシステムにおけるシミュレーションの実行と同様に達成できる。図 1 のシステムにおいて、ユーザは、制御モジュール 24 によって、またはその内部に既に与えられているシミュレーション能力に依存してその入力および出力パラメータを設定した。例えば、デルタ V 制御装置システムにおいて、ユーザは、制御装置またはフィールド装置のために設計された処理制御モジュール 24 において入力/出力機能ブロックのシミュレートパラメータを手動で設定するか、またはコンピュータ 40 内の制御モジュール 24 の入力/出力ブロックとインタフェースして、これらの制御モジュールによって生成された出力にตอบสนองし、制御モジュール 24 に対する入力を生成する特定のシミュレーション処理制御モジュールを提供することも可能である。その代わりに、またはそれに関連して、シミュレーションアプリケーション 32 はコンピュータ 40 内で実行されて制御モジュール 24 と相互作用し、実際の処理機能をシミュレートすることも可能である。この場合、処理制御モジュール 24 またはユーザインタフェース 27 内の入力/出力シミュレーションパラメータ等のアクセスパラメータは、周知の OPC インタフェースを利用して支持可能であり、それによって、ワークステーション 18 (例えば、表示アプリケーション 26)、構成データベース 30、制御装置 12 (例えば、制御装置アプリケーション 23) または OPC インタフェースを使用するフィールド装置において通常実行されるアプリケーションを実行し、チェックアウトすることが可能になる。例えば、HYSYS 製品等のシミュレーションソフトウェアとデルタ V 制御装置アプリケーション等の制御装置アプリケーションとの間で通信する OPC インタフェースを開発することも可能である。当然のことながら、他のインタフェースおよびシミュレーションアプリケーションも同様に使用可能である。さらに、制御装置アプリケーション 23 をコンピュータ 40 に取り付けることにより、例えば、C++ または他のプログラミング環境によって提供されるデバッグツールを用いて、制御モジュール 24 またはユーザインタフェース 27 等のコンピュータ 40 上のいずれかのソフトウェアを障害追及するために使用することも可能である。

20

30

40

【0022】

明らかに、制御装置アプリケーション 23 (およびその処理制御モジュール 24) は、現在、図 1 に示すような外部データハイウェイを介さずに、コンピュータ 40 内で表示アプリケーション 26、シミュレーションアプリケーション 36、データベース構成アプリケーション 32 および先回り処理制御および診断アプリケーション等の他の望ましいアプリ

50

ケーションと通信する。

【0023】

図2の統合システムを使用すると、分散型処理制御システムにおける異なるハードウェア装置内で記憶され実行される異なるアプリケーションおよびモジュールが単一独立型コンピュータにおいて実現され、例えば、処理制御モジュール24やユーザインタフェース27の構成もしくはテストを可能にするか、または制御装置12、フィールド装置15もしくは構成データベース30等の専用ハードウェアを必要とすることなくシミュレーション環境内でユーザまたはオペレータの訓練を可能にすることもできる。この方法において、これらの構成要素に関連する設計、テストおよび訓練は、専用の分散型処理制御システムハードウェアが購入されるか、または入手可能になるまでに実施可能である。さらに、
10 処理制御モジュール24およびユーザインタフェース26は、一旦テストされると、これ以上変更することなく、制御装置、フィールド装置（例えば、フィールドバスフィールド装置）または実際の分散型処理システムの実行時に使用する異なるワークステーションに割り当て、ダウンロードすることが可能である。その理由は、好ましい実施の形態において、これらの構成要素が、いずれかの装置で実行され、異なる装置の構成要素と通信できる通信インタフェースを有するオープンオペレーティングシステムで実行するように設計されているからである。このように、これらのソフトウェア構成要素は、制御装置12またはフィールド装置15等の専用ハードウェアの特定の部分での実行に限定されない。これは、いずれかのオープンプラットフォームにおいて実行され、他のハードウェア装置に
20 における他のソフトウェア構成要素と通信するように最初に設計されたデルタV制御装置ソフトウェアの場合に特に当てはまり、特定のベンダに関連する特定のハードウェアで実行されるとは限らない。

【0024】

処理制御ソフトウェアを単一PCまたは他のコンピュータ上の表示ソフトウェアおよびシミュレーションソフトウェアと統合する上記アプローチには、構成開発、制御装置およびオペレータインタフェースチェックアウト、およびオペレータ訓練に必要なハードウェアを劇的に減じるという利点がある。特に、分散型制御システムのために開発されている処理制御モジュール24は、実際の制御装置が存在しない状態で専用制御装置またはフィールド装置内にあると仮定してコンピュータ40に割り当て、実行することも可能である。実際、処理制御モジュール24は、入力/出力機能ブロックのシミュレーション能力を
30 用いて、コンピュータ40内で作成され、記憶され、実行されて、処理をシミュレートし、それによって、専用ハードウェアを購入するか、または取り付けるまでに制御システムチェックアウトおよびオペレータ訓練を可能にすることもできる。

【0025】

制御装置アプリケーション23はコンピュータ40によって実行され、処理の際、実際のフィールド装置に接続されないか、またはそれを制御しないため、処理制御モジュール24が実行時に専用制御装置12またはフィールド装置15のうちの1個で実行される実時間速度以外の速度で実行できるように制御装置アプリケーション23を設定または構成可能である。例えば、デルタVシステムにおいて、制御モジュール24の各々は、実行時におけるその実行速度、すなわち一定期間に制御モジュールが何度実行されるべきかを定めるパラメータを備える。しかし、制御装置アプリケーション23は、コンピュータ40
40 において実行される際、実時間速度よりも速く、または遅く処理制御モジュール（またはその機能ブロック）を実行するように構成可能である。その理由は、処理制御動作がコンピュータ40においてシミュレートされるのみであり、実際には実施されないからである。

【0026】

一般に、処理制御モジュールを実時間よりも速く実行することによって、訓練シミュレーションおよびいくつかの設計アプリケーションに利点が得られる。例えば、複合処理において、設定点や他のオペレータ制御パラメータへの変更によって、長時間にわたって処理に著しい影響を与えることはない。これらの場合、シミュレートされた応答時間を訓練
50

および設計のために、例えば1分または2分早めることが望ましい。例えばデバッグ手順において処理制御モジュール24の実行速度を落として、設計者がより遅い速度でモジュールの内部作業をみることができるようになることが望ましい。同様に、高忠実度シミュレーションプログラムは、CPU42のかなりの処理電力(時間)を費やす可能性があるため、特に制御装置アプリケーション23および表示アプリケーション26と共に実時間速度で処理制御モジュール24を実行することが不可能である。これらの場合、制御モジュール24の実行速度を低減して、完全シミュレーションを可能にすることが望ましい。

【0027】

当然のことながら、制御アプリケーションによってユーザは、制御モジュール24が実行される速度を規定もしくは選択するか、または処理制御モジュール24内でブロックをいかに速く実行するかを決定する際に制御装置アプリケーションによって使用する実行速度乗数、例えば、2倍、5倍、3分の1等を規定することが可能である。制御装置アプリケーションが専用制御装置12またはフィールド装置15によって実行される時、実時間よりも速いか、または遅い制御モジュールを実行する能力が自動的に不能化されて、処理の実行時に誤った実行速度の使用を防止するのが好ましい。当選のことながら、他の形式の制御装置アプリケーションは、他の方法で実時間実行速度を規定することが可能であるため、制御装置アプリケーションの特定の構成によって言及される他のいずれかの望ましい技術を用いてパーソナルコンピュータ40における実行速度を変えることができる。

【0028】

さらに、コンピュータ40に記憶された制御装置アプリケーション23は、既知の分散型制御システム内で他の制御プラットフォームとインタフェースする先回り制御プラットフォームを提供するために使用可能である。例えば、図3を参照すると、コンピュータ40、特に制御装置アプリケーション23は、通信回線48(データハイウェイであってもよい)を介して制御装置50に接続可能であり、この制御装置50は、異なる型式の制御装置であってもよく、また異なる処理制御装置製造業者によって提供されるような異なる制御装置アプリケーションを実行するものであってもよい。図3に示すように、制御装置50は、制御装置50によって生成されるデータを表示し、制御装置50内で処理制御モジュールまたは他のソフトウェアを変更するためのアプリケーションと、制御装置50によって実行される制御装置ソフトウェアに関連する他のアプリケーションとを実行するワークステーション52に通信回線48を介して接続可能である。同様に、制御装置50は、処理内で実際のフィールド装置(図示せず)に接続され、それを制御することも可能である。

【0029】

図3の制御装置アプリケーション23は、制御装置50と通信して、望ましいデータを獲得し、またコマンドを送って制御装置50の動作を変更する場合もある。このように、コンピュータ40における制御装置、表示、データベース構成または他のアプリケーション、例えば診断および傾向アプリケーションの高度な機能性を、他のベンダからの制御装置アプリケーションと共に使用することも可能である。

【0030】

この先回り制御を可能にするために、制御装置アプリケーション23は、構成アプリケーション23と制御装置50において実行する外部システムとの間でインタフェースするように特別に開発されたOPCインタフェースを使用することも可能である。あるいは、構成アプリケーション23は、制御装置50に結合されたデータヒストリアンを介して通信することを一般に必要とするPI等の履歴インタフェースアプリケーションの能力を利用して外部または異なる制御装置システムにインタフェースすることも可能である。当然のことながら、他の望ましいインタフェースも同様に使用可能である。制御装置アプリケーション23および制御装置50の両方がOPCインタフェースを支持する好ましい実施の形態では、フィッシャ・ローズマウント・システムズ・インク(Fisher Rosemount Systems, Inc.)の提供するOPCミラーシェアウェアを用い

10

20

30

40

50

てOPCサーバを共に結合し、それによって、例えば回線48上で2個のOPCインタフェース間で通信を行うことも可能である。

【0031】

いずれの場合でも、陰機能ブロックとして知られている特定の制御モジュールを、コンピュータ40の処理制御モジュール24内に与えることも可能であり、アプリケーション23が使用するプロトコルと制御装置50が使用するプロトコルとの間で制御パラメータまたは値をマッピングするために使用可能である。一般に、陰機能ブロックは、制御装置アプリケーション23が使用するプロトコル内で動作して、そのシステム内の他の機能ブロックにデータを送り、またそこからコマンドおよびデータを受け取る機能ブロックである。実際の機能を行う代わりに、陰機能ブロックは、外部機能ブロックまたは処理制御装置50内の他のソフトウェアエレメントの状態をミラー化し、それに対するインタフェースとして動作する。特に、陰機能ブロックは、OPC、PIまたは他のインタフェースを用いて、処理制御装置50と定期的に通信し、制御装置50によって制御される処理制御システム内で実効値を反映するデータまたはパラメータ値を獲得し、その後、制御装置アプリケーション23が使用するプロトコルを介して制御装置アプリケーション23内で他の機能ブロックまたは処理制御モジュールがアクセス可能か、または利用可能となるようにこのデータを記憶することも可能である。陰機能ブロックもまた、制御装置アプリケーション23が実行する他の機能ブロックからコマンドまたはデータを受信し、OPC、PIまたは他のインタフェースを用いて、対応する外部機能ブロックまたは処理制御装置50内の他のソフトウェアエレメントにこれらのコマンドまたはデータを送る。このように、陰機能ブロックは、制御装置50が制御する制御システム内で論理装置の現在の状態を反映し、コンピュータ40内の制御装置アプリケーション23が生成するコマンドおよび他のデータを制御装置50に送り、それによって、そのシステムを変更する。陰機能ブロックの実現に関する詳細は、本発明の主題ではないが、ここで引用している、本発明の譲受人に譲渡された1998年9月10日出願の「処理制御ネットワークにおいて使用するための陰機能ブロックインタフェース」と題する米国特許出願第09/151,084号に詳細に述べられている。

【0032】

いずれの場合でも、データは、一旦インタフェースを介して制御装置アプリケーション23に与えられると、処理モジュール24のいずれか、表示アプリケーション26および対応するユーザインタフェース27、または他のアプリケーション、例えば制御装置アプリケーション23に関連する拡張アプリケーションによって使用することができる。これらの拡張アプリケーションは、いずれかの形態、例えば、モデル予測制御アプリケーション、ニューラルネットワークアプリケーション、ならびに本発明の譲受人に譲渡され、ここで引用している1999年2月22日出願の「処理制御システムにおける診断」と題する米国特許出願第09/256,585号に述べられた診断アプリケーションのような警報、検査および診断アプリケーションであってもよい。制御システムの標準ソフトウェアを先回り制御およびモニタを実現するための他の制御システムに階層化する能力は、処理制御産業にとって新しい能力である。

【0033】

ここで述べたソフトウェアのいずれかは、実現されると、コンピュータまたはプロセッサ等のRAMまたはROMにおける磁気ディスク、レーザーディスクまたは他の記憶媒体等のコンピュータ読み出し可能メモリに記憶することも可能である。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ディスクもしくは他の転送可能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、他のローカル・エリア・ネットワークもしくはワイド・エリア・ネットワーク等の通信チャネルを含む既知の、または望ましい配信方法によってユーザ、処理制御システムまたはコンピュータ40に配信可能である(この配信は、転送可能記憶媒体によるこのようなソフトウェアの提供と同じか、または交換可能なものとして見なされる)。さらに、このソフトウェアは、変調しないで直接与えてもよく、また通信チャネル上で送信される前に、好適な変調搬

10

20

30

40

50

送波を用いて変調することも可能である。また、ワークステーション、コンピュータ、ラップトップ等の用語はすべて、ここでは交換可能に用いられ、いずれかの形態の処理またはコンピュータ装置を示す。

【 0 0 3 4 】

このように、本発明は、例示するのみであり、本発明を限定しない特定の実施例を参照して述べたが、本発明の精神および範囲を逸脱することなく開示した実施の形態に変更、追加または削除を行っても良いことが当業者に明らかとなるであろう。

【 図面の簡単な説明 】

【 0 0 3 5 】

【 図 1 】異なるハードウェア装置によって異なる処理制御機能が実施される典型的な処理制御ネットワークのブロック図である。 10

【 図 2 】図 1 と同様に、分散型処理制御環境において構成設計、システムチェックアウトおよび処理制御アクティビティのシミュレーションのための異なる分散型処理制御アプリケーションを実行する単一コンピュータのブロック図である。

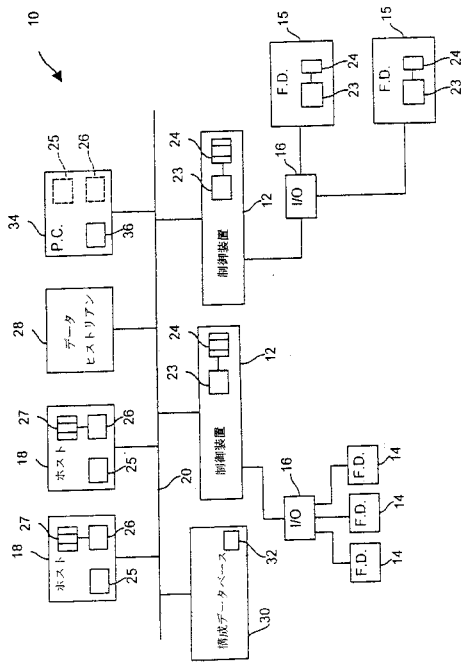
【 図 3 】第 2 の制御装置アプリケーションを実行する制御装置に通信結合された図 2 のコンピュータのブロック図である。

【 符号の説明 】

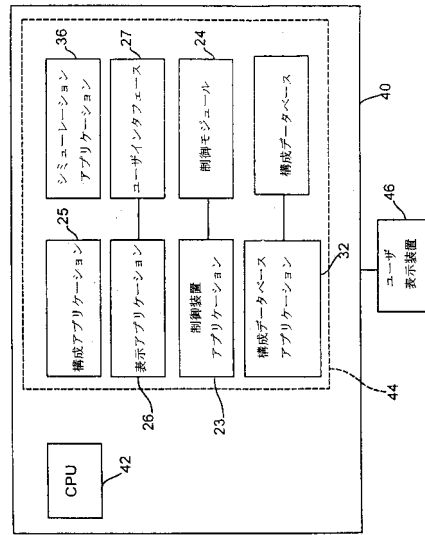
【 0 0 3 6 】

1 2	処理制御装置	
1 5	フィールド装置	20
1 6	入力 / 出力モジュール	
1 8 , 5 2	ワークステーション	
2 0	データハイウェイ	
2 4	制御アプリケーション	
2 5	構成アプリケーション	
2 6	表示アプリケーション	
2 8	データヒストリアン	
3 0	構成データベース	
3 2	構成データベースアプリケーション	
3 6	シミュレーションアプリケーション	30
4 0 , 5 0	コンピュータ	
4 2	C P U	
4 4	メモリ	
4 8	通信回線	

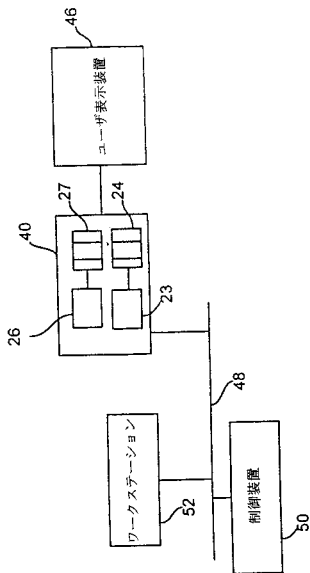
【図1】



【図2】



【図3】



フロントページの続き

(72)発明者 テレンス エル． ブレピンス

アメリカ合衆国 78681 テキサス ラウンド ロック カーメル ドライブ 3801

(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

審査官 桑原 雅子

(56)参考文献 米国特許第05752008 (US, A)

(58)調査した分野(Int.Cl., DB名)

G06F 15/00

G06F 13/00

G05B 23/02