



(10) DE 10 2011 052 840 A1 2012.02.23

(12)

Offenlegungsschrift

(21) Aktenzeichen: 10 2011 052 840.7

(51) Int Cl.: G06F 11/28 (2011.01)

(22) Anmeldetag: 19.08.2011

G05B 19/04 (2011.01)

(43) Offenlegungstag: 23.02.2012

G05B 23/00 (2011.01)

(30) Unionspriorität:

US 12/861, 515

23.08.2010 US

(74) Vertreter:

Meissner, Bolte & Partner GbR, 80538, München,
DE

(71) Anmelder:

Fisher-Rosemount Systems, Inc., Austin, Tex., US

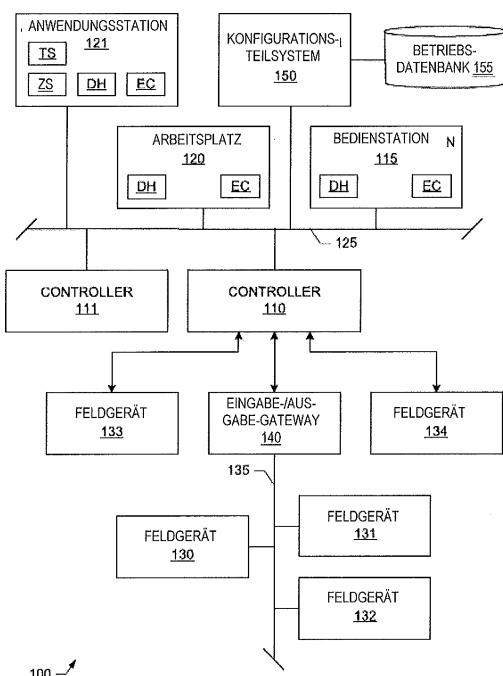
(72) Erfinder:

Aneweer, Tom, Georgetown, Tex., US; Bell, Noel,
Austin, Tex., US; Thiele, Dirk, Austin, Tex., US

Die folgenden Angaben sind den vom Anmelder eingereichten Unterlagen entnommen

(54) Bezeichnung: Verfahren, Vorrichtungen und Fertigungsartikel zum Testen von Prozesssteuersystemen

(57) Zusammenfassung: Es werden beispielhafte Verfahren, Vorrichtungen und Fertigungsartikel zum Testen von Prozesssteuersystemen offenbart. Ein offbarte beispielhaftes Verfahren beinhaltet das Erhalten von Nutzereingaben in ein erstes Prozesssteuersystem, das Erhalten von Prozesseingaben und ersten Prozessausgaben des ersten Prozesssteuersystems, das Liefern der Nutzereingaben und der Prozesseingaben zu einem zweiten Prozesssteuersystem, um das zweite Prozesssteuersystem zu betreiben, das Erhalten zweiter Prozessausgaben des zweiten Prozesssteuersystems, das mit den Nutzereingaben und den Prozesseingaben betrieben wird, und das Vergleichen der ersten und der zweiten Prozessausgaben, um festzustellen, ob das Prozesssteuersystem wie vorgesehen implementiert ist.



Beschreibung**GEBIET DER ERFINDUNG**

[0001] Die vorliegende Offenbarung betrifft allgemein Prozesssteuersysteme und insbesondere Verfahren, Vorrichtungen und Fertigungsartikel zum Testen von Prozesssteuersystemen.

STAND DER TECHNIK

[0002] Während des Lebenszyklus einer Verarbeitungsanlage muss unter Umständen das Prozesssteuersystem ausgetauscht, repariert, korrigiert, höhergestuft usw. werden. Jede Veränderung eines Prozesssteuersystems hat jedoch das Potenzial, den Betrieb der Verarbeitungsanlage zu stören.

KURZDARSTELLUNG

[0003] Es werden beispielhafte Verfahren, Vorrichtungen und Artikel zum Testen von Prozesssteuersystemen offenbart. Ein offenbartes beispielhaftes Verfahren beinhaltet das Erhalten von Nutzereingaben in ein erstes Prozesssteuersystem, das Erhalten von Prozesseingaben und ersten Prozessausgaben des ersten Prozesssteuersystems, das Liefern der Nutzereingaben und der Prozesseingaben zu einem zweiten Prozesssteuersystem, um das zweite Prozesssteuersystem zu betreiben, das Erhalten zweiter Prozessausgaben des zweiten Prozesssteuersystems, das mit den Nutzereingaben und den Prozesseingaben betrieben wird, und das Vergleichen der ersten und der zweiten Prozessausgaben, um festzustellen, ob das zweite Prozesssteuersystem wie vorgesehen implementiert ist.

[0004] Offenbare beispielhafte maschinenlesbare Befehle bewirken bei der Ausführung, dass ein Prozessor Nutzereingaben in ein erstes Prozesssteuersystem erhält, Prozesseingaben und erste Prozessausgaben des ersten Prozesssteuersystems erhält, die Nutzereingaben und die Prozesseingaben zu einem zweiten Prozesssteuersystem liefert, um das zweite Prozesssteuersystem zu betreiben, zweite Prozessausgaben des zweiten Prozesssteuersystems, das mit den Nutzereingaben und den Prozesseingaben betrieben wird, erhält, die Nutzereingaben und die Prozesseingaben zu einem dritten Prozesssteuersystem liefert, um das dritte Prozesssteuersystem zu betreiben, dritte Prozessausgaben des dritten Prozesssteuersystems, das mit den Nutzereingaben und den Prozesseingaben betrieben wird, erhält, und die zweiten und dritten Prozessausgaben vergleicht, um zu bestimmen, ob das dritte Prozesssteuersystem wie vorgesehen implementiert ist.

[0005] Eine offenbare beispielhafte Vorrichtung weist einen Datensammler zum Erhalten von Nutzereingaben, Prozesseingaben und ersten Prozess-

ausgaben eines ersten Prozesssteuersystems und zum Erhalten zweiter Prozessausgaben eines zweiten Prozesssteuersystems auf, einen Wiedergeber zum Liefern der Nutzereingaben und der Prozesseingaben zum zweiten Prozesssteuersystem, um das zweite Prozesssteuersystem zu betreiben, und einen Vergleicher zum Vergleichen der ersten und zweiten Prozessausgaben, um festzustellen, ob das zweite Prozesssteuersystem wie vorgesehen gelaufen ist.

KURZBESCHREIBUNG DER ZEICHNUNGEN

[0006] [Fig. 1](#), [Fig. 2](#) und [Fig. 3](#) sind schematische Abbildungen beispielhafter Prozesssteuersysteme, die gemäß den Lehren der vorliegenden Offenbarung aufgebaut sind.

[0007] [Fig. 4](#) bildet eine beispielhafte Art und Weise der Implementierung des beispielhaften Testservers von [Fig. 1](#) bis [Fig. 3](#) ab.

[0008] [Fig. 5](#) und [Fig. 6](#) sind Ablaufdiagramme, die beispielhafte Prozesse darstellen, die ausgeführt werden können, um einen oder alle der beispielhaften Testserver von [Fig. 1](#) bis [Fig. 4](#) zu implementieren und/oder um Prozesssteuersysteme zu testen.

[0009] [Fig. 7](#) ist eine schematische Abbildung einer beispielhaften Prozessorplattform, die dafür verwendet und/oder programmiert werden kann, die beispielhaften Prozesse von [Fig. 5](#) und/oder [Fig. 6](#) auszuführen und/oder einen oder alle der beispielhaften Testserver von [Fig. 1](#) bis [Fig. 4](#) zu implementieren.

AUSFÜHLICHE BESCHREIBUNG

[0010] Aufgrund der Komplexität moderner Prozesssteuersysteme und der grenzenlosen Zahl von Verarbeitungsanlagenkonfigurationen kann es schwierig sein, alle Mängel zu erkennen, bevor eine Reparatur (Fix), Korrektur (Patch), höhere Version (Upgrade) usw. eines Prozesssteuersystems freigegeben wird. Solche unerkannten Mängel treten unter Umständen erst nach und/oder bei Betrieb des Prozesssteuersystems in Verbindung mit einer realen Verarbeitungsanlage zu Tage. Um die Notwendigkeit des Austauschens, Modifizierens, Korrigierens, Aktualisierens und/oder Höherstufens ihres Prozesssteuersystems/ihrer Prozesssteuersysteme zu verringern, haben sich einige Kunden stattdessen entschieden, ihr/e Prozesssteuersystem/e von anderen Geräten und/oder Netzwerken zu isolieren. Zwar können solche Maßnahmen das Potenzial für Störungen verringern, doch hindert es auch Prozesstechniker daran, die Merkmale und/oder Funktionen, die in neueren Versionen eines Prozesssteuersystems zur Verfügung stehen, auszunutzen.

[0011] Allgemein können die hierin beschriebenen beispielhaften Vorrichtungen, Verfahren und Ferti-

gungsartikel verwendet werden, um ein neues und/oder aktualisiertes Prozesssteuersystem unter Verwendung von Prozessdaten zu testen, die von einer realen Verarbeitungsanlage eines Kunden unter Verwendung des vorhandenen Prozesssteuersystems bzw. der vorhandenen Prozesssteuersysteme gesammelt werden, bevor das neue und/oder aktualisierte Prozesssteuersystem zum Steuern der Verarbeitungsanlage des Kunden verwendet wird. Insbesondere werden Nutzereingaben (die z. B. über grafische Nutzerschnittstellen empfangen wurden), Prozesseingaben (die z. B. von Feldgeräten empfangen wurden) und Prozessausgaben (die z. B. zu Feldgeräten gesendet wurden) eines ersten Prozesssteuersystems erfasst, während das erste Prozesssteuersystem innerhalb der Verarbeitungsanlage des Kunden läuft. Die erfassten Eingaben werden verwendet, um ein zweites Prozesssteuersystem in Gang zu setzen und/oder zu betreiben, während seine Prozessausgaben erfasst werden. Das heißt, die erfassten Eingaben vom ersten Prozesssteuersystem werden angewandt, um das zweite Prozesssteuersystem zu betreiben, statt dass das zweite Prozesssteuersystem als Reaktion auf Nutzereingaben und/oder Eingaben, die von Feldgeräten empfangen wurden, betrieben wird.

[0012] Mit den hierin beschriebenen Verfahren, Vorrangordnungen und Fertigungsartikeln ist es nicht erforderlich, ein Modell der Verarbeitungsanlage zu definieren und/oder zu implementieren, um das zweite Prozesssteuersystem zu testen. Stattdessen können die von den beiden Prozesssteuersystemen erfassten Prozessausgaben verglichen werden, um festzustellen, ob das zweite Prozesssteuersystem wie vorgesehen gelaufen ist. Da die Daten erfasst wurden, während das erste Prozesssteuersystem innerhalb der Verarbeitungsanlage des Kunden lief, können die Prozesstechniker, falls die erfassten Prozessausgaben übereinstimmen, sicher sein, dass das zweite Prozesssteuersystem wie vorgesehen in der Verarbeitungsanlage des Kunden laufen wird. Wegen der begrenzten Genauigkeit (z. B. Quantisierung von Zeit und/oder Quantisierung numerischer Werte) stimmen die erfassten Prozessausgaben unter Umständen nicht genau überein, sollten jedoch in ausreichendem Maße übereinstimmen, um eine Feststellung zu ermöglichen, dass das zweite Prozesssteuersystem wie vorgesehen gelaufen ist.

[0013] Das zweite Prozesssteuersystem kann das erste Prozesssteuersystem nach Anwendung einer Reparatur, Korrektur, einem Upgrade usw. sein, jedoch bevor das erste Prozesssteuersystem mit dem Steuern der Verarbeitungsanlage beginnt. Zusätzlich oder alternativ kann das zweite Prozesssteuersystem in einer anderen Umgebung und/oder an einem anderen physischen Standort implementiert sein, wie etwa in einer unternehmenseigenen Testeinrichtung und/oder in einer Test- oder Schulungseinrichtung, die

zum Lieferanten der Anlage gehört. Ferner können das erste und/oder das zweite Prozesssteuersystem eine Simulation und/oder ein Modell eines Prozesssteuersystems sein.

[0014] In einigen Beispielen werden die erfassten Eingaben auch verwendet, um ein drittes Prozesssteuersystem in Gang zu setzen und/oder zu betreiben, während seine Prozessausgaben erfasst werden. Das heißt, die erfassten Eingaben vom ersten Prozesssteuersystem werden angewandt, um das zweite sowie auch das dritte Prozesssteuersystem zu betreiben. Die Prozessausgaben des zweiten und dritten Prozesssteuersystems können verglichen werden, um festzustellen, ob das dritte Prozesssteuersystem wie vorgesehen gelaufen ist. Wenn zum Beispiel das zweite Prozesssteuersystem identisch mit dem ersten Prozesssteuersystem ist und sich nur insofern unterscheidet, als das zweite Prozesssteuersystem separat von der Verarbeitungsanlage des Kunden betrieben wird, und das dritte Prozesssteuersystem das zweite Prozesssteuersystem nach Anwendung einer Reparatur, Korrektur, eines Upgrades usw. ist, können die Prozesstechniker durch Vergleichen der Prozessausgaben des zweiten und dritten Prozesssteuersystems sicher sein, dass das aktualisierte Prozesssteuersystem in der Verarbeitungsanlage des Kunden wie vorgesehen laufen wird.

[0015] Die vom ersten Prozesssteuersystem erfassten Daten können zusätzlich oder alternativ auch verwendet werden, um ein oder mehrere Szenarien zu erkennen und/oder zu ermitteln, die in der Verarbeitungsanlage wiederholt auftreten können. Zum Beispiel können die vom ersten Prozesssteuersystem erfassten ursprünglichen Eingaben und Ausgaben mit zusätzlichen, vom ersten Prozesssteuersystem erfassten Eingaben und Ausgaben verglichen werden, statt die erfassten Eingaben auf ein zweites Prozesssteuersystem anzuwenden. Die ursprünglichen und zusätzlichen Eingaben und Ausgaben können verglichen werden, um wiederholt auftretende Szenarien zu erkennen und/oder zu ermitteln. Wenn zum Beispiel die ursprünglichen und zusätzlichen Eingaben und Ausgaben übereinstimmen, möglicherweise innerhalb gewisser Toleranzen, kann ein Bediener des Prozesssteuersystems benachrichtigt werden, dass das mit den ursprünglichen Eingaben und Ausgaben einhergehende Szenario in der Verarbeitungsanlage aufgetreten ist und/oder derzeit auftritt.

[0016] Weiterhin können die hierin beschriebenen Beispiele zum Implementieren von Schulungssystemen verwendet werden. Zum Beispiel können die vom ersten Prozesssteuersystem erfassten Eingaben und Ausgaben verwendet werden, um das zweite Prozesssteuersystem zu betreiben, während eine zu schulende Person den Betrieb des zweiten Prozesssteuersystems beobachtet und/oder das zweite Prozesssteuersystem als Reaktion auf die vorher erfass-

ten Eingaben betreibt. Im letzteren Fall können die Eingaben der zu schulenden Person mit den erfassenen Nutzereingaben verglichen werden, um zu bestimmen, ob die zu schulende Person angemessen und/oder unverzüglich reagiert hat.

[0017] [Fig. 1](#) ist eine schematische Abbildung eines beispielhaften Prozesssteuersystems **100**, das verwendet werden kann, um beispielsweise einen Fertigungsprozess, ein Fertigungssystem, eine Fertigungsanlage und/oder jede andere Art von Verarbeitungsanlage ganz oder teilweise zu steuern. Im Interesse der Kürze und Klarheit wird in allen folgenden Beschreibungen auf das beispielhafte Prozesssteuersystem **100** von [Fig. 1](#) verwiesen. Die hierin beschriebenen Verfahren, Vorrichtungen und Fertigungsartikel zum Testen von Prozesssteuersystemen sind jedoch auch auf andere Prozesssteuersysteme anwendbar. Das beispielhafte Prozesssteuersystem **100** von [Fig. 1](#) weist einen oder mehrere Prozesscontroller (von denen zwei mit den Bezugszeichen **110** und **111** bezeichnet sind), eine oder mehrere Bedienstationen (von denen eine mit dem Bezugszeichen **115** bezeichnet ist) und eine oder mehrere Arbeitsplätze (von denen zwei mit den Bezugszeichen **120** und **121** bezeichnet sind) auf. Die beispielhaften Prozesscontroller **110** und **111**, die beispielhafte Bedienstation **115** und die beispielhaften Arbeitsplätze **120** und **121** sind über einen Bus und/oder ein lokales Netzwerk (LAN) **125** kommunikativ verbunden, was für gewöhnlich als Anwendungssteuerernetzwerk (ACN, Application Control Network) bezeichnet wird.

[0018] Die beispielhafte Bedienstation **115** von [Fig. 1](#) gestattet es einem Bediener der Verarbeitungsanlage, eine oder mehrere Bedienanzeigebildschirme, grafische Nutzerschnittstellen und/oder Anwendungen zu prüfen und/oder zu bedienen, die es dem Bediener der Verarbeitungsanlage ermöglichen, Verarbeitungsanlagenvariablen anzusehen, Verarbeitungsanlagenzustände anzusehen, Verarbeitungsanlagenbedingungen anzusehen, Verarbeitungsanlagenalarme anzusehen und/oder Nutzereingaben bereitzustellen wie beispielsweise jene, die verwendet werden, um Verarbeitungsanlageneinstellungen zu ändern (z. B. Sollwerte und/oder Betriebszustände, hörbare Alarne, stumme Alarne usw.). Solche Bildschirme und/oder Anwendungen werden üblicherweise von Prozesskonfigurationstechnikern konzipiert und/oder implementiert.

[0019] Die beispielhaften Arbeitsplätze **120** und **121** von [Fig. 1](#) können dafür konfiguriert sein, jede beliebige Anzahl und/oder Art(en) von Anwendung(en) und/oder Funktion(en) zu implementieren. Im abgebildeten Beispiel von [Fig. 1](#) ist der Arbeitsplatz **120** dafür konfiguriert, hauptsächlich auf die Prozesssteuerung bezogene Anwendungen auszuführen, während der beispielhafte Arbeitsplatz **121** dafür konfiguriert ist,

hauptsächlich Kommunikationsanwendungen auszuführen, die es dem Prozesssteuersystem **100** ermöglichen, mit anderen Geräten oder Systemen unter Verwendung beliebiger gewünschten Kommunikationsmedien (z. B. drahtlos, fest verdrahtet usw.) und -protokolle (z. B. HTTP, SOAP usw.) zu kommunizieren. Zum Beispiel kann der beispielhafte Arbeitsplatz **121** eine oder mehrere Anwendungen der Informationstechnologie, nutzerinteraktive Anwendungen und/oder Kommunikationsanwendungen implementieren. Der Arbeitsplatz **121** wird im Folgenden als die Anwendungsstation **121** bezeichnet. Wie im Weiteren ausführlicher beschrieben wird, implementiert und/oder weist die beispielhafte Anwendungsstation **121** von [Fig. 1](#) einen Testserver TS auf, der verwendet werden kann, um ein Prozesssteuersystem wie etwa jedes beliebige der Prozesssteuersysteme **100**, **200** und **300** von [Fig. 1](#) bis [Fig. 3](#) zu testen. Eine beispielhafte Art und Weise der Implementierung des beispielhaften Testservers ist im Weiteren im Zusammenhang mit [Fig. 4](#) beschrieben. Beispieldhafte Prozesse, die zum Beispiel vom beispielhaften Testserver TS ausgeführt werden können, um ein Prozesssteuersystem zu testen, sind im Weiteren im Zusammenhang mit [Fig. 5](#) und [Fig. 6](#) beschrieben.

[0020] Die beispielhaften Stationen **115**, **120** und **121** können unter Verwendung eines beliebigen geeigneten Computersystems und/oder Verarbeitungssystems wie etwa der beispielhaften Prozessorplattform P100 von [Fig. 7](#) implementiert sein. Die Stationen **115**, **120** und **121** können zum Beispiel unter Verwendung von Einzelprozessor- und/oder Mehrprozessoren-Computern implementiert sein.

[0021] Das beispielhafte LAN **125** von [Fig. 1](#) kann unter Verwendung eines beliebigen gewünschten Kommunikationsmediums oder -protokolls implementiert sein. Zum Beispiel kann das LAN **125** auf einem verdrahteten und/oder drahtlosen Ethernet-Kommunikationsmodell basieren. Es können jedoch auch beliebige andere geeignete Kommunikationsmedien und/oder -protokolle verwendet werden. Ferner ist zwar in [Fig. 1](#) ein einziges LAN **125** abgebildet, doch kann auch mehr als ein LAN und/oder andere alternative Kommunikationshardware verwendet werden, um redundante Kommunikationspfade innerhalb des beispielhaften Prozesssteuersystems **100** von [Fig. 1](#) bereitzustellen.

[0022] Der beispielhafte Controller **110** von [Fig. 1](#) ist über einen digitalen Datenbus **135** und ein Eingabe-/Ausgabe-Gateway **140** mit mehreren intelligenten Feldgeräten **130**, **131** und **132** verbunden. Die intelligenten Feldgeräte **130** bis **132** können feldbusfähige Ventile, Stellantriebe, Sensoren usw. sein. In diesem Fall kommunizieren die intelligenten Feldgeräte **130** bis **132** über den digitalen Datenbus **135** unter Verwendung des allgemein bekannten Foundation-Fieldbus-Protokolls. Natürlich können stattdessen

auch andere Arten intelligenter Feldgeräte und Kommunikationsprotokolle verwendet werden. Zum Beispiel könnten die intelligenten Feldgeräte **130** bis **132** stattdessen Profibus- und/oder HART-fähige Geräte sein, die unter Verwendung des allgemein bekannten Profibus- und HART-Kommunikationsprotokolle über den Datenbus **135** kommunizieren. Mit dem Controller **110** können zusätzliche Eingabe-/Ausgabegeräte gekoppelt sein, die sich vom Eingabe-/Ausgabe-Gateway **140** unterscheiden, diesem ähneln oder mit diesem identisch sind, um zusätzlichen Gruppen von intelligenten Feldgeräten, die Foundation-Fieldbus-Geräte, HART-Geräte usw. sein können, die Kommunikation mit dem Controller **110** zu ermöglichen. Zusätzlich zu den beispielhaften intelligenten Feldgeräten **130** bis **132** können ein oder mehrere nicht-intelligenten Feldgeräte **133** und **134** kommunikativ mit dem beispielhaften Controller **110** verbunden sein. Die beispielhaften nicht-intelligenten Feldgeräte **133** und **134** von [Fig. 1](#) können zum Beispiel herkömmliche Geräte mit 4–20 mA oder 0–10 V Gleichstrom (VDC) sein, die über entsprechende Links mit den Controllern **110** und **111** kommunizieren. Zwar ist dies in [Fig. 1](#) nicht dargestellt, doch ist der beispielhafte Controller **111** ebenfalls mit ähnlichen und/oder anderen Feldgeräten und/oder Eingabe-/Ausgabe-Gateways verbunden.

[0023] Die beispielhaften Controller **110** und **111** von [Fig. 1](#) können zum Beispiel ein DeltaV™-Controller und/oder eine DeltaV-Characterization-Module-(Charms-)Eingabe-/Ausgabekarte sein, die von Fisher-Rosemount Systems, Inc., einem Unternehmen der Emerson Process Management, vertrieben wird. Es können auch beliebige andere Controller verwendet werden. Ferner sind zwar in [Fig. 1](#) zwei Controller **110** und **111** dargestellt, doch können auch zusätzliche Controller einer beliebigen gewünschten Art und/oder eine beliebige Kombination von Arten mit dem LAN **125** gekoppelt sein. Jedenfalls implementieren und/oder führen die beispielhaften Controller **110** und **111** eine oder mehrere zum Prozesssteuerungssystem **100** gehörende Prozesssteuerroutinen und/oder -module aus, die von einem Systemtechniker und/oder einem anderen Systembediener unter Verwendung der Bedienstation **115** erzeugt wurden und die in die Controller **110** und **111** heruntergeladen und/oder instanziert wurden.

[0024] Zum Konfigurieren der Steuerkomponenten des beispielhaften Prozesssteuersystems **100** (z. B. der beispielhaften Controller **110** und **111**, der beispielhaften Bedienstation **115**, der beispielhaften Arbeitsplätze **120** und **121**, des beispielhaften Eingabe-/Ausgabe-Gateways **140** und/oder der beispielhaften Feldgeräte **130** bis **134**) weist das beispielhafte Prozesssteuersystem **100** von [Fig. 1](#) ein Konfigurationsteilsystem **150** des Prozesssteuersystems auf. Das beispielhafte Konfigurationsteilsystem **150** von [Fig. 1](#) lädt, konfiguriert, programmiert die realen (d.

h. physischen) Prozesssteuerkomponenten des Prozesssteuersystems **100** auf der Basis einer Betriebsdatenbank **155** und/oder nimmt diese Komponenten in Betrieb. Das beispielhafte Konfigurationsteilsystem **150** und die beispielhafte Betriebsdatenbank **155** von [Fig. 1](#) sind Teil des DeltaV-Prozesssteuersystems, das von Fisher-Rosemount Systems, Inc., einem Unternehmen der Emerson Process Management, vertrieben wird. Die beispielhafte Betriebsdatenbank **155** von [Fig. 1](#) kann auch mit anderen Tools und/oder Schnittstellen (nicht dargestellt) des DeltaV-Prozesssteuersystems erzeugt, definiert, spezifiziert und/oder bestückt sein.

[0025] Zum Erfassen, Aufzeichnen und/oder Protokollieren von Nutzer- oder Bedienereingaben weist die beispielhafte Bedienstation **115** von [Fig. 1](#) einen Ereignis-Chronisten EC auf. Wenn ein Nutzer mit der beispielhaften Bedienstation **115** interagiert, um zum Beispiel Sollwerte oder Prozessvariablen bereitzustellen und/oder zu ändern, eine Anwendung zu starten, eine grafische Nutzerschnittstelle zu öffnen und/oder zu schließen usw., erfasst und/oder zeichnet der beispielhafte Ereignis-Chronist EC von [Fig. 1](#) die Eingaben des Nutzers in einem Ereignisprotokoll auf. Der beispielhafte Ereignis-Chronist EC von [Fig. 1](#) kann zum Beispiel der DeltaV Event Chronicle sein, der von Fisher-Rosemount Systems, Inc., einem Unternehmen der Emerson Process Management, vertrieben wird. Wie in [Fig. 1](#) dargestellt, können auch der beispielhafte Arbeitsplatz **120** und die beispielhafte Anwendungsstation **121** einen Event-Chronisten EC mit im Wesentlichen ähnlicher Funktionalität implementieren.

[0026] Zum Erfassen, Aufzeichnen und/oder Protokollieren von Prozesseingaben und Prozesssteuerausgaben implementieren und/oder weisen eine/r oder mehrere der beispielhaften Bedienstation **115**, des beispielhaften Arbeitsplatzes **120** und/oder der beispielhaften Anwendungsstation **121** einen Data Historian DH auf. Die beispielhaften Data Historians DH von [Fig. 1](#) erfassen und/oder zeichnen die an den beispielhaften Controllern **110** und **111** empfangenen Prozesseingaben in einem Datenprotokoll auf. Die Data Historians DH zeichnen auch von den beispielhaften Controllern **110** und **111** erzeugte Prozesssteuerausgaben in dem Datenprotokoll auf. Die beispielhaften Data Historians DH von [Fig. 1](#) können zum Beispiel durch den DeltaV Continuous Historian implementiert sein, der von Fisher-Rosemount Systems, Inc., einem Unternehmen der Emerson Process Management, vertrieben wird.

[0027] Zwar werden die Ereignis-Chronisten EC und Data Historian DH im abgebildeten Beispiel von [Fig. 1](#) zum Erfassen von Eingaben und Ausgaben des beispielhaften Prozesssteuersystems **100** verwendet, doch könnte zusätzlich oder alternativ auch ein redundanter, zweiter und/oder Reserve-Control-

ler (nicht dargestellt) zum Erfassen der Eingaben und/oder Ausgaben verwendet werden. Der redundante Controller könnte dafür konfiguriert sein, seinen Zustand mit einem überwachten Controller **110**, **111** zu synchronisieren und Daten, Befehle und/oder Informationen aufzuzeichnen, die zum und/oder vom überwachten Controller **110**, **111** gesendet wurden. Die aufgezeichneten Informationen könnten später vom redundanten Controller abgerufen und, wie im Weiteren beschrieben, vom beispielhaften Testserver TS zum Testen eines Prozesssteuersystems verwendet werden. Der Zustand des redundanten Controllers könnte auch verwendet werden, um den Zustand eines zu testenden Prozesssteuersystems zu initialisieren, bevor es mit aufgezeichneten Nutzer- und Prozesseingaben in Gang gesetzt wird. In einigen Beispielen können die vom redundanten Controller erfassten Daten mit geringerer Genauigkeit erfasst (z. B. quantisiert) werden, um die Datenmenge zu verringern, die zwischen den überwachten und den redundanten Controllern übertragen werden muss. Wenn erfasste Prozessausgaben später verglichen werden, können Unterschiede hinsichtlich des anfänglichen Zustands und der anfänglichen Daten aufgrund der Quantisierung berücksichtigt werden. Der redundante Controller kann die Daten erfassen, während er im Prozesssteuersystem eines Kunden installiert ist, und später zwecks Analyse, Schulung und/oder Test eines anderen Prozesssteuersystems in ein Labor überführt werden.

[0028] Zum Zugreifen auf Nutzereingaben, Prozesseingaben und/oder Prozessausgaben, die von den beispielhaften Ereignis-Chronisten EC und/oder den beispielhaften Data Historians DH erfasst wurden, implementiert und/oder weist die beispielhafte Anwendungsstation **121** von [Fig. 1](#) einen Zugangsserver ZS auf. Der beispielhafte Zugangsserver ZS von [Fig. 1](#) greift unter Verwendung einer beliebigen Anzahl und/oder Art von Verfahren, die gemäß einer vergangenen, gegenwärtigen und/oder zukünftigen Norm und/oder Spezifikation zur Objektverknüpfung und Einbettung (OLE, Object Linking and Embedding) zur Prozesssteuerung (OPC) implementiert sind, auf die protokollierten Nutzereingaben, Prozesseingaben und Prozessausgaben zu. Der beispielhafte Zugangsserver ZS von [Fig. 1](#) ermöglicht es anderen Elementen des Prozesssteuersystems **100** wie etwa dem beispielhaften Testserver TS, auf protokollierte Nutzereingaben, Prozesseingaben und/oder Prozessausgaben zuzugreifen. Der beispielhafte Zugangsserver ZS ermöglicht es dem Testserver TS auch, den Controllern **110** und **111** Nutzereingaben und/oder Prozesseingaben zu liefern, um zum Beispiel zu bewirken, dass die beispielhaften Controller **110** und **111** von [Fig. 1](#) in Abwesenheit der Feldgeräte **130** bis **134** und/oder des Eingabe-/Ausgabe-Gateways **140** und/oder sogar in Abwesenheit einer Verarbeitungsanlage zu laufen. Der beispielhafte Zugangsserver ZS von [Fig. 1](#) kann bei-

spielsweise durch den DeltaV History Server und/oder dem DeltaV OPC Events Server implementiert sein, die von Fisher-Rosemount Systems, Inc., einem Unternehmen der Emerson Process Management, vertrieben werden.

[0029] Zum Testen eines Prozesssteuersystems wie etwa eines beliebigen der beispielhaften Prozesssteuersysteme **100**, **200** und **300** von [Fig. 1](#), [Fig. 2](#) bzw. [Fig. 3](#) weist die beispielhafte Anwendungsstation **121** von [Fig. 1](#) den beispielhaften Testserver TS auf. Während das beispielhafte Prozesssteuersystem **100** von [Fig. 1](#) läuft, sammeln die beispielhaften Ereignis-Chronisten EC und die beispielhaften Data Historians DH Eingaben und Ausgaben der Controller **110** und **111**. Der beispielhafte Testserver TS von [Fig. 1](#) erhält über den beispielhaften Zugangsserver ZS die gesammelten Eingaben und Ausgaben zur späteren Verwendung im Betrieb des beispielhaften Prozesssteuersystems **100** von [Fig. 1](#) und/oder anderer Prozesssteuersysteme wie etwa derer, die in [Fig. 2](#) und [Fig. 3](#) dargestellt sind. Zum Testen eines Prozesssteuersystems betreibt der beispielhafte Testserver TS das zu testende Prozesssteuersystem unter Verwendung der aufgezeichneten Eingaben und Ausgaben und/oder setzt es in Gang. Das heißt, die erfassten Eingaben werden direkt auf das zu testende Prozesssteuersystem angewandt, um das zu testende Prozesssteuersystem zu betreiben, statt dass das zu testende Prozesssteuersystem als Reaktion auf Nutzereingaben und/oder Eingaben läuft, die von der Bedienstation **115**, den Feldgeräten **130** bis **134** und/oder dem Eingabe-/Ausgabe-Gateway **140** empfangen werden. Der Testserver TS wendet die erfassten Eingaben über den Zugangsserver ZS auf das zu testende Prozesssteuersystem an.

[0030] Während der beispielhafte Testserver TS von [Fig. 1](#) das zu testende Prozesssteuersystem betreibt, sammeln die Ereignis-Chronisten EC und die Data Historians DH Nutzereingaben und Prozesseingaben und/oder -ausgaben. Der beispielhafte Testserver TS kann die von den beiden Prozesssteuersystemen erfassten Prozessausgaben vergleichen, um festzustellen, ob das zweite Prozesssteuersystem wie vorgesehen gelaufen ist. Da die Daten erfasst wurden, während das erste Prozesssteuersystem in der Verarbeitungsanlage des Kunden lief, kann der Testserver TS und/oder ein Nutzer des Testservers TS, wenn die erfassten Prozessausgaben übereinstimmen, mit Sicherheit feststellen, dass das zu testende Prozesssteuersystem in einer Verarbeitungsanlage des Kunden wie vorgesehen laufen sollte. Wegen der begrenzten Genauigkeit (z. B. Quantisierung von Zeit und/oder numerischen Werten) stimmen die erfassten Prozessausgaben unter Umständen nicht genau überein, sollten jedoch in ausreichendem Maße übereinstimmen, um eine Feststellung zu ermöglichen, dass das zu testende Prozesssteuersystem wie vorgesehen gelaufen ist. Zum Beispiel können

analoge Werte als übereinstimmend betrachtet werden, wenn ihre Differenz einen vom Nutzer definierbaren Schwellenwert nicht überschreitet, und/oder digitale Werte können als übereinstimmend betrachtet werden, wenn sie mit einem vom Nutzer definierbaren Fenster in Bezug aufeinander erscheinen.

[0031] [Fig. 2](#) bildet ein beispielhaftes zu testendes Prozesssteuersystem **200** ab, das zum Testen des beispielhaften Prozesssteuersystems **100** von [Fig. 1](#) verwendet werden kann. Da viele der in [Fig. 2](#) abgebildeten Elemente identisch mit denen sind, die im Vorangegangenen im Zusammenhang mit [Fig. 1](#) erläutert wurden, werden identische Elemente in [Fig. 1](#) und [Fig. 2](#) mit identischen alphanumerischen Bezeichnungen bezeichnet, und zu einer Beschreibung identischer Elemente wird der Leser auf die Beschreibungen im Vorangegangenen im Zusammenhang mit [Fig. 1](#) verwiesen.

[0032] In dem abgebildeten beispielhaften zu testenden Prozesssteuersystem **200** von [Fig. 2](#) wurden die beispielhaften Controller **110** und **111** logisch und/oder physisch von den beispielhaften Feldgeräten **130** bis **134** und/oder dem Eingabe-/Ausgabe-Gateway **140** getrennt. Dementsprechend laufen die beispielhaften Controller **110** und **111** von [Fig. 2](#) als Reaktion auf vorher erfasste Prozesseingaben und Nutzereingaben, die über den Zugangsserver ZS vom Testserver TS an die Controller **110** und **111** geliefert werden. Wie in [Fig. 2](#) dargestellt, ist keine reale Verarbeitungsanlage zum Testen des Prozesssteuersystems **200** erforderlich. Ferner ist auch kein Modell der Verarbeitungsanlage erforderlich. Stattdessen werden vorher erfasste Prozesseingaben in das zu testende Prozesssteuersystem **200** wiedergegeben und die Controller **110** und **111** berechnen Prozesssteuerausgaben, als würden sie in einer realen Prozesssteueranlage laufen. Der beispielhafte Testserver TS wendet die Eingaben unabhängig von den vom Prozesssteuersystem **200** berechneten Ausgaben auf das Prozesssteuersystem **200** an. Wenn also die ursprünglich erfassten Eingaben einen stabilen Betrieb des Prozesssteuersystems **100** darstellen, so sollten die Ausgaben des Prozesssteuersystems **200** gleichermaßen stabil sein.

[0033] Es sollte offensichtlich sein, dass das beispielhafte Prozesssteuersystem **200** von [Fig. 2](#) getestet werden kann, noch bevor die Betriebsdatenbank **155** Eingabe-/Ausgabebezüge enthält, wie zum Beispiel in einer frühen Phase der Vorbereitung der Prozesssteuerung. In solchen Beispielen können vom beispielhaften Testserver TS Eingaben und Ausgaben angewandt werden, die von einer ähnlichen Anlage erfasst wurden, und/oder Eingaben, die zum Testen des Prozesssteuersystems **200** konzipiert, modifiziert und/oder vorgesehen sind.

[0034] Da die beispielhaften Controller **110** und **111** keine angeschlossenen Feldgeräte **130** bis **134** und/oder keinen angeschlossenen Eingabe-/Ausgabe-Gateway **140** haben, muss die beispielhafte Betriebsdatenbank **155** des Prozesssteuersystems **200** so modifiziert sein, dass sie die echten Eingabe-/Ausgabe-Kanalbelegungsinformationen durch logische und/oder simulierende Eingabeparameter ersetzt. Der Ersatz der Eingabe-/Ausgabekanalbelegungsinformationen kann unter Verwendung der Anwendung DeltaV SimulatePro erfolgen, die von Fisher-Rosemount Systems, Inc., einem Unternehmen der Emerson Process Management, vertrieben wird.

[0035] In einigen Beispielen ist der Betriebsdatenbank **155** ein Gegenmodul (nicht dargestellt) hinzugefügt, um die Abfrage des Betriebs des zu testenden Prozesssteuersystems **200** in gewünschten Zeitintervallen und/oder mit einer bestimmten Rate zu ermöglichen. Eine beispielhafte Abfragerate ist eine Sekunde, obwohl auch jede andere Rate implementiert sein kann, die von den Datenübertragungsfunktionen des Zugangsservers ZS unterstützt wird. In einigen Beispielen sind auch alle Prozesssteuerroutinen, die mit einer höheren Rate als die Abfragerate laufen, so modifiziert, dass sie mit der Abfragerate laufen. Alternativ können gepufferte Lese- und/oder gepufferte Schreibvorgänge vom Zugangsserver ZS implementiert sein, um es Prozesssteuerroutinen zu ermöglichen, bei Raten zu laufen, die die Abfragerate übersteigen.

[0036] Das beispielhafte Prozesssteuersystem **200** von [Fig. 1](#) kann beispielsweise das Prozesssteuersystem **100** nach Anwendung einer Reparatur, Korrektur, eines Upgrades usw. sein, jedoch bevor das Prozesssteuersystem **200** mit dem Steuern der Verarbeitungsanlage beginnt. Zusätzlich oder alternativ kann das Prozesssteuersystem **200** in einer anderen Umgebung und/oder an einem anderen physischen Standort implementiert sein, wie etwa in einer unternehmenseigenen Testeinrichtung und/oder in einer Test- oder Schulungseinrichtung, die zum Lieferanten des Prozesssteuersystems **200** gehört. Ferner kann, wie in [Fig. 3](#) dargestellt ist, eine Simulation und/oder ein Modell eines Prozesssteuersystems verwendet werden, um das beispielhafte Prozesssteuersystem **100** zu implementieren, und/oder es kann eine Simulation und/oder ein Modell einer Verarbeitungsanlage verwendet werden, um das beispielhafte Prozesssteuersystem **100** zu testen.

[0037] [Fig. 3](#) bildet ein weiteres beispielhaftes zu testendes Prozesssteuersystem **300** ab, das zum Testen des beispielhaften Prozesssteuersystems **100** verwendet werden kann. Da viele der in [Fig. 3](#) abgebildeten Elemente identisch mit denen sind, die im Vorangegangenen im Zusammenhang mit [Fig. 1](#) erläutert wurden, werden identische Elemente in [Fig. 1](#) und [Fig. 3](#) mit identischen alphanumerischen Be-

zugszeichen bezeichnet, und zu einer Beschreibung identischer Elemente wird der Leser auf die Beschreibungen im Vorangegangenen im Zusammenhang mit [Fig. 1](#) verwiesen.

[0038] Im abgebildeten beispielhaften zu testenden Prozesssteuersystem **300** von [Fig. 3](#) sind die beispielhaften Controller **110** und **111** durch einen Simulator SIM ersetzt und/oder von diesem modelliert und/oder simuliert. In dem beispielhaften Prozesssteuersystem **300** von [Fig. 3](#) ist der beispielhafte Simulator SIM auf der beispielhaften Betriebsdatenbank **115** implementiert. Der Simulator SM kann jedoch alternativ auch auf der beispielhaften Anwendungsstation **121**, auf dem beispielhaften Arbeitsplatz **120** und/oder auf jedem anderen Verarbeitungs- oder Computersystem implementiert sein. Der beispielhafte Simulator SIM von [Fig. 3](#) kann unter Verwendung der Anwendung DeltaV SimulatePro implementiert sein, die von Fisher-Rosemount Systems, Inc., einem Unternehmen der Emerson Process Management, vertrieben wird.

[0039] Zwar bilden [Fig. 1](#), [Fig. 2](#) und [Fig. 3](#) beispielhafte Prozesssteuersysteme **100**, **200** bzw. **300** ab, innerhalb derer die beispielhaften Vorrichtungen, Verfahren und Fertigungsartikel zum Testen von Prozesssteuersystemen vorteilhaft eingesetzt werden können, doch wird ein Durchschnittsfachmann ohne Weiteres erkennen, dass die hierin beschriebenen Vorrichtungen, Verfahren und Fertigungsartikel nach Wunsch auch vorteilhaft in anderen Verarbeitungsanlagen und/oder Prozesssteuersystemen größerer oder geringerer Komplexität (z. B. mit mehr als zwei Controllern, über mehr als einen geografischen Standort usw.) als die abgebildeten Beispiele von [Fig. 1](#) bis [Fig. 3](#) eingesetzt werden können. Darüber hinaus kann jede Anzahl und/oder Art(en) von zusätzlichen und/oder alternativen Geräten, Komponenten und/oder Systemen in einer Verarbeitungsanlage und/oder einem Prozesssteuersystem enthalten sein, obschon dies in [Fig. 1](#) bis [Fig. 3](#) der Klarheit der Veranschaulichung halber nicht dargestellt ist. Zum Beispiel kann eine Verarbeitungsanlage und/oder ein Prozesssteuersystem eine Firewall, einen Schalter, einen Router, einen Hub, eine Energiezufuhr und/oder beliebige weitere Geräte aufweisen und/oder implementieren, die von einem Prozesssteuersystem verwaltet und/oder gesteuert werden können, wie beispielsweise vom DeltaV-Prozesssteuersystem, das von Fisher-Rosemount Systems, Inc., einem Unternehmen der Emerson Process Management, vertrieben wird.

[0040] Zwar sind in [Fig. 1](#), [Fig. 2](#) und [Fig. 3](#) beispielhafte Prozesssteuersysteme **100**, **200** bzw. **300** abgebildet, doch können eine/s oder mehrere der in [Fig. 1](#), [Fig. 2](#) und [Fig. 3](#) abgebildeten Schnittstellen, Datenstrukturen, Elemente, Prozesse und/oder Geräte auch auf jede beliebige andere Art und Weise

kombiniert, getrennt, neu angeordnet, weggelassen, beseitigt und/oder implementiert sein. Ferner können die beispielhaften Controller **110**, **111**, die beispielhaften Stationen/Plätze **115**, **120** und **121**, die beispielhaften Feldgeräte **130** bis **134**, das beispielhafte Eingabe-/Ausgabe-Gateway **140**, das beispielhafte Konfigurationsteilsystem **150**, die beispielhaften Ereignis-Chronisten EC, die beispielhaften Data Historians DH, der beispielhafte Zugangsserver ZS und/oder der beispielhafte Testserver TS mit Hardware, Software, Firmware und/oder jeder beliebigen Kombination aus Hardware, Software und/oder Firmware implementiert sein. So kann zum Beispiel jede/r/s der beispielhaften Controller **110**, **111**, der beispielhaften Stationen/Plätze **115**, **120** und **121**, der beispielhaften Feldgeräte **130** bis **134**, des beispielhaften Eingabe-/Ausgabe-Gateways **140**, des beispielhaften Konfigurationsteilsystems **150**, der beispielhaften Ereignis-Chronisten EC, der beispielhaften Data Historians DH, des beispielhaften Zugangsservers ZS und/oder des beispielhaften Testservers TS mit der beispielhaften Prozessorplattform P100 von [Fig. 7](#) und/oder mit einem oder mehreren Schaltkreisen, programmierbaren Prozessoren, anwendungsspezifischen Schaltkreisen (ASICs), programmierbaren logischen Schaltungen (PLDs), feldprogrammierbaren logischen Schaltungen (FPLDs), feldprogrammierbaren Gate-Arrays (FPGAs), Sicherungen usw. implementiert sein. Wird ein Vorrichtungsanspruch des vorliegenden Patents, der eines oder mehrere dieser Elemente aufweist, dahingehend ausgelegt, dass er auch eine reine Software- und/oder Firmware-Implementierung beinhaltet, so wird hiermit mindestens eine/r/s der beispielhaften Controller **110**, **111**, der beispielhaften Stationen/Plätze **115**, **120** und **121**, der beispielhaften Feldgeräte **130** bis **134**, des beispielhaften Eingabe-/Ausgabe-Gateways **140**, des beispielhaften Konfigurationsteilsystems **150**, der beispielhaften Ereignis-Chronisten EC, der beispielhaften Data Historians DH, des beispielhaften Zugangsservers ZS und/oder des beispielhaften Testservers TS ausdrücklich dahingehend definiert, dass er/sie es einen physischen Fertigungsartikel aufweist, wie beispielweise ein physisches computerlesbares Medium, das die Firmware und/oder Software speichert. Außerdem können die beispielhaften Prozesssteuersysteme **100**, **200** und/oder **300** Schnittstellen, Datenstrukturen, Elemente, Prozesse und/oder Geräte statt jener oder zusätzlich zu jenen aufweisen, die in [Fig. 1](#), [Fig. 2](#) und/oder [Fig. 3](#) abgebildet sind, und/oder sie können mehr als eine/n/s von einem oder allen der abgebildeten Schnittstellen, Datenstrukturen, Elemente, Prozesse und/oder Geräte aufweisen.

[0041] Im Rahmen des Vorliegenden ist der Begriff „physisches computerlesbares Medium“ ausdrücklich dahingehend definiert, dass es jede Art von computerlesbarem Medium beinhaltet und weitergeleitete Signale ausdrücklich ausschließt. Im Rahmen des Vorliegenden ist der Begriff „nicht-transito-

risches computerlesbares Medium" ausdrücklich dahingehend definiert, dass es jede Art von computerlesbarem Medium beinhaltet und weitergeleitete Signale ausdrücklich ausschließt. Beispielhafte physische und/oder nicht-transitorische computerlesbare Medien sind u. a. ein flüchtiger und/oder nicht-flüchtiger Speicher, ein flüchtiges und/oder nicht-flüchtiges Speichergerät, eine Compact Disc (CD), eine Digital Versatile Disc (DVD), eine Diskette, eine Lesespeicher (ROM), ein Speicher mit wahlfreiem Zugriff (RAM), ein programmierbarer ROM (PROM), ein elektronisch programmierbarer ROM (EPROM), ein elektronisch löscherbarer PROM (EEPROM), ein optisches Speicherlaufwerk, ein optisches Speichergerät, ein magnetisches Speicherlaufwerk, ein magnetisches Speichergerät, ein Cachespeicher und/oder jedes andere Speichermedium, in dem Informationen über einen beliebigen Zeitraum (z. B. für längere Zeiträume, permanent, für kurze Zeit, für zeitweiliges Puffer und/oder zum Zwischenspeichern der Informationen) gespeichert sind und auf das mit einem Prozessor, einem Computer und/oder einer anderen Maschine mit einem Prozessor, wie mit der beispielhaften Prozessorplattform P100, die im Weiteren im Zusammenhang mit [Fig. 7](#) erläutert wird, zugegriffen werden kann.

[0042] [Fig. 4](#) bildet eine beispielhafte Art und Weise der Implementierung eines beliebigen der beispielhaften Testserver TS von [Fig. 1](#), [Fig. 2](#) und [Fig. 3](#) ab. Zum Sammeln von Daten weist der beispielhafte Testserver TS von [Fig. 4](#) einen Datensammler **405** auf. Der beispielhafte Datensammler **405** von [Fig. 4](#) interagiert mit dem beispielhaften Zugangsserver ZS ([Fig. 1](#) bis [Fig. 3](#)), um Nutzereingaben, Prozesseingaben und/oder Prozessausgaben von einem Prozesssteuersystem zu erhalten. Der beispielhafte Datensammler **405** speichert die erfassten Eingaben und Ausgaben unter Verwendung einer beliebigen Anzahl und/oder Art(en) von Datenstruktur(en), Datei(en) und/oder Dateiformat(en) in einer Datenbank **410**. In einigen Beispielen interagiert der Datensammler **405** mit dem beispielhaften Zugangsserver ZS, um die erfassten Eingaben und Ausgaben im Microsoft®-Excel®-Format zu erhalten. Die beispielhafte Datenbank **410** kann auf jeder beliebigen Anzahl und Art(en) von flüchtigen und/oder nicht-flüchtigen Speichergeräten und/oder Speichern gespeichert sein.

[0043] Zum Erzeugen von Daten, die zum In-Gang-Setzen und/oder Betreiben eines Prozesssteuersystems geeignet sind, weist der beispielhafte Testserver TS einen Datenmodifizierer **415** auf. Der beispielhafte Datenmodifizierer **415** kombiniert Nutzereingabeinformationen, die von den Ereignis-Chronisten EC gesammelt wurden, mit Prozesseingaben, die von den Data Historians DH gesammelt wurden, zu einer zeitindizierten Datenstruktur, wobei jeder Eintrag der Datenstruktur einem Abfragezeitintervall entspricht. Die erzeugte zeitindizierte Datenstruktur kann in der

beispielhaften Datenbank **410** gespeichert werden. In einigen Beispielen werden Microsoft Excel zur Implementierung des beispielhaften Datenmodifizierers **415** und eine Microsoft-Excel-Datei zum Implementieren der zeitindizierten Datenstruktur verwendet. Es können jedoch auch beliebige andere Datenstrukturen verwendet werden.

[0044] Einige von den Data Historians DH erfasste Eingaben liegen unter Umständen nicht in dem gleichen Format vor, in dem sie von den Controllern **110** und **111** empfangen wurden. Zum Beispiel könnte ein Analogeingabe-(AI-)Block eines Proportional-Integral-Derivative-(PID-)Steuerkreises eine Prozessvariante wie etwa den Druck auf einen Wert skalieren oder in einen Wert konvertieren, der einen Tankfüllstand darstellt. In solchen Fällen skaliert der beispielhafte Datenmodifizierer **415** von [Fig. 4](#) die erfasste(n) Prozesseingabe(n), so dass diese skalierte(n) Prozesseingaben(n) die ursprüngliche Eingabe darstellen. Zum Beispiel könnte ein Wert für den Tankfüllstand so skaliert werden, dass er einen Druckwert darstellt. In einigen Beispielen wird das Skalieren der erfassten Prozesseingaben unter Verwendung einer Funktion zum gebündelten Überarbeiten in einer Anwendung wie Microsoft Excel implementiert. Zum Beispiel können Skalierungsdaten von der DeltaV-Konfiguration **155** erhalten und in einer Microsoft-Excel-Datei gespeichert werden. Es können maschinenzugängliche Befehle, die zum Beispiel als Microsoft-Visual-Basic-(VBA-)Code implementiert sind, zum Ausführen der Skalierung der erfassten Prozesseingaben verwendet werden.

[0045] Zusätzlich oder alternativ können einige erfasste Prozesseingaben durch „benannte Mengen“ dargestellt werden. Zum Beispiel kann ein Wert von 1 „offen“ oder „geschlossen“ darstellen, je nachdem, ob das dazugehörige Ventil „normal offen“ oder „normal geschlossen“ ist. In solchen Fällen ersetzt der beispielhafte Datenmodifizierer **415** von [Fig. 4](#) den Text der benannten Menge (z. B. „geschlossen“), der von den Ereignis-Chronisten EC und/oder den Data Historians DH erfasst wurde, durch den entsprechenden numerischen Wert (z. B. „1“). In einigen Beispielen wird die Ersetzung von Text benannter Mengen durch numerische Werte unter Verwendung einer Funktion zum gebündelten Überarbeiten in einer Anwendung wie Microsoft Excel implementiert. Die Zuordnungen von Text benannter Mengen zu numerischen Werten können zum Beispiel durch das Exportieren von Informationen benannter Mengen aus dem DeltaV Explorer, der von Fisher-Rosemount Systems, Inc., einem Unternehmen der Emerson Process Management, vertrieben wird, in zum Beispiel eine Microsoft-Excel-Datei erhalten werden. Es können maschinenzugängliche Befehle, die zum Beispiel als Microsoft-Visual-Basic-(VBA-)Code implementiert sind, zum Ersetzen von Text benannter Mengen durch numerische Werte verwendet werden.

[0046] In einigen Beispielen komprimiert der beispielhafte Datenmodifizierer **415** die zeitindizierte Datenstruktur, um die Menge des erforderlichen Speichers zu verringern. Zum Beispiel ist es nicht ungewöhnlich, dass Daten, die von einer realen Verarbeitungsanlage erfasst werden, Zeitintervalle aufweisen, in denen keine Nutzereingaben erfolgen und/oder sich keine Prozesseingaben ändern. Der beispielhafte Datenmodifizierer **415** von [Fig. 4](#) ersetzt fortlaufende Einträge, in denen Änderungen erfolgten, durch einen einzigen Eintrag, der einen Zählerwert der fortlaufenden Einträge darstellt. In einigen Beispielen werden nur Folgen unveränderter Eingaben mit Längen ersetzt, die einen vom Nutzer definierbaren Schwellenwert überschreitet.

[0047] Zum In-Gang-Setzen und/oder Betreiben eines Prozesssteuersystems weist der beispielhafte Testserver TS von [Fig. 4](#) einen Wiedergeber **420** auf. Der beispielhafte Wiedergeber **420** von [Fig. 4](#) sendet jeden vom Datenmodifizierer **415** erzeugten Eingabeeintrag der zeitindizierten Datenstruktur über den beispielhaften Zugangsserver ZS zu den/dem entsprechenden logischen und/oder simulierten Eingabe-/Ausgabe-Port(s). Die Eingaben werden an den Zeitmarken geliefert, die zu den Eingabeeinträgen gehören. Während der beispielhafte Wiedergeber **420** von [Fig. 4](#) das Prozesssteuersystem in Gang setzt, sammeln die beispielhaften Ereignis-Chronisten EC und die Data Historians DH Nutzereingaben, Prozesseingaben und Prozessausgaben.

[0048] Zur Feststellung, ob ein Prozesssteuersystem wie vorgesehen gelaufen ist, weist der beispielhafte Testserver TS von [Fig. 4](#) einen Vergleicher **425** auf. Der beispielhafte Vergleicher **425** von [Fig. 4](#) vergleicht die ursprünglich erfassten Prozessausgaben mit den Prozessausgaben, die erfasst wurden, während der Wiedergeber **420** das Prozesssteuersystem in Gang setzt, um zum Beispiel festzustellen, ob ein zu testendes Prozesssteuersystem wie vorgesehen gelaufen ist. Wegen der begrenzten Genauigkeit (z. B. Quantisierung von Zeit und/oder numerischen Werten) stimmen die erfassten Prozessausgaben unter Umständen nicht genau überein und so verwendet der Vergleicher **425** beim Vergleichen von Prozessausgabenwerten eine oder mehrere Toleranzen.

[0049] Zusätzlich oder alternativ kann der Vergleicher **425** verwendet werden, um ein oder mehrere Szenarien zu erkennen und/oder zu ermitteln, die in einer Verarbeitungsanlage wiederholt auftreten können. Zum Beispiel kann der Vergleicher **425** vorher erfasste Eingaben und Ausgaben mit zusätzlichen erfassten Eingaben und Ausgaben vergleichen, die zum Erkennen und/oder Ermitteln wiederholt auftretender Szenarien erfasst wurden, statt dass der Wiedergeber **420** die erfassten Eingaben auf ein Prozesssteuersystem anwendet. Wenn zum Beispiel die ursprünglichen und die zusätzlichen Eingaben

und Ausgaben übereinstimmen, möglicherweise innerhalb gewisser Toleranzen, kann ein Bediener des Prozesssteuersystems benachrichtigt werden, dass das mit den ursprünglichen Eingaben und Ausgaben einhergehende Szenario in der Verarbeitungsanlage aufgetreten ist und/oder derzeit auftritt.

[0050] Außerdem kann der Vergleicher **425** zum Implementieren von Schulungssystemen verwendet werden. Zum Beispiel können die von einem ersten Prozesssteuersystem erfassten Eingaben und Ausgaben verwendet werden, um ein zweites Prozesssteuersystem zu betreiben, während eine zu schulende Person den Betrieb des zweiten Prozesssteuersystems beobachtet und/oder das zweite Prozesssteuersystem als Reaktion auf die vorher erfassten Ausgaben betreibt. Im letzteren Fall vergleicht der Vergleicher **425** die Eingaben der zu schulenden Person mit den erfassten Nutzereingaben, um festzustellen, ob die zu schulende Person angemessen und/oder unverzüglich reagiert hat.

[0051] Zwar ist in [Fig. 4](#) eine beispielhafte Art und Weise der Implementierung der beispielhaften Testserver TS von [Fig. 1](#), [Fig. 2](#) und/oder [Fig. 3](#) abgebildet, doch können ein oder mehrere der in [Fig. 4](#) abgebildeten Elemente, Prozesse und/oder Geräte auch auf jede beliebige andere Art und Weise kombiniert, getrennt, neu angeordnet, weggelassen, besetzt und/oder implementiert sein. Ferner können der beispielhafte Datensammler **405**, die beispielhafte Datenbank **410**, der beispielhafte Datenmodifizierer **415**, der beispielhafte Wiedergeber **420**, der beispielhafte Vergleicher **425** und/oder allgemeiner der beispielhafte Testserver TS von [Fig. 4](#) durch Hardware, Software, Firmware und/oder jede beliebige Kombination aus Hardware, Software und/oder Firmware implementiert sein. So kann zum Beispiel jede/r des beispielhaften Datensammlers **405**, der beispielhaften Datenbank **410**, des beispielhaften Datenmodifizierers **415**, des beispielhaften Wiedergebers **420**, des beispielhaften Vergleichers **425** und/oder allgemeiner des beispielhaften Testservers TS mit der beispielhaften Prozessorplattform P100 von [Fig. 7](#) und/oder mit einem oder mehreren Schaltkreisen, programmierbaren Prozessoren, ASICs, PLDs, FPLDs, FPGAs, Sicherungen usw. implementiert sein. Wird ein Vorrichtungsanspruch des vorliegenden Patents, der eines oder mehrere dieser Elemente aufweist, dahingehend ausgelegt, dass er auch eine reine Software- und/oder Firmware-Implementierung beinhaltet, so wird hiermit mindestens eine/r des beispielhaften Datensammlers **405**, der beispielhaften Datenbank **410**, des beispielhaften Datenmodifizierers **415**, des beispielhaften Wiedergebers **420**, des beispielhaften Vergleichers **425** und/oder allgemeiner des beispielhaften Testservers TS ausdrücklich dahingehend definiert, dass er/sie einen physischen Fertigungsartikel aufweist, wie beispielweise ein physisches computerlesbares Medium, das die Firmware

und/oder Software speichert. Außerdem kann der beispielhafte Testserver TS von [Fig. 4](#) ein oder mehr Elemente, Prozesse und/oder Geräte statt jener oder zusätzlich zu jenen aufweisen, die in [Fig. 4](#) abgebildet sind, und/oder er kann mehr als einen/s von einem oder allen der abgebildeten Elemente, Prozesse und/oder Geräte aufweisen.

[0052] [Fig. 5](#) und [Fig. 6](#) sind Ablaufdiagramme, die beispielhafte Prozesse darstellen, die ausgeführt werden können, um einen oder alle der beispielhaften Testserver TS von [Fig. 1](#), [Fig. 2](#), [Fig. 3](#) und/oder [Fig. 4](#) zu implementieren. Ein Prozessor, ein Controller und/oder ein beliebiges anderes Verarbeitungsgerät können dafür verwendet, konfiguriert und/oder programmiert werden, die beispielhaften Prozesse von [Fig. 5](#) und/oder [Fig. 6](#) auszuführen. Zum Beispiel können die beispielhaften Prozesse von [Fig. 5](#) und [Fig. 6](#) in codierten oder maschinenlesbaren Befehlen verkörpert sein, die auf einem physischen computerlesbaren Medium gespeichert sind. Maschinenlesbare Befehle umfassen zum Beispiel Befehle, die bewirken, dass ein Prozessor, ein Computer und/oder eine Maschine mit einem Prozessor einen oder mehrere konkrete Prozesse ausführen. Alternativ können einige oder alle der beispielhaften Prozesse von [Fig. 5](#) und/oder [Fig. 6](#) mit einer beliebigen Kombination aus ASICs, PLDs, FPLDs, PFGAs, diskreter Schaltung, Hardware, Firmware usw. implementiert werden. Auch können einige oder alle der beispielhaften Prozesse von [Fig. 5](#) und/oder [Fig. 6](#) manuell oder als beliebige Kombination aus einer der genannten Techniken implementiert werden, zum Beispiel als Kombination aus Firmware, Software, diskreter Schaltung und/oder Hardware. Ferner können auch viele andere Verfahren zur Implementierung der beispielhaften Operationen von [Fig. 5](#) und/oder [Fig. 6](#) eingesetzt werden. Zum Beispiel kann die Reihenfolge der Ausführung der Blöcke geändert werden und/oder einer oder mehrere der beschriebenen Blöcke kann geändert, weggelassen, weiter unterteilt oder kombiniert werden. Zusätzlich können die Blöcke eines oder aller der beispielhaften Prozesse von [Fig. 5](#) und/oder [Fig. 6](#) nacheinander und/oder parallel ausgeführt werden, zum Beispiel durch separate Verarbeitungsthreads, Prozessoren, Geräte, diskrete Schaltungen, Schaltkreise usw.

[0053] Der beispielhafte Prozess von [Fig. 5](#) kann ausgeführt werden, um Daten zur Verwendung beim Testen eines Prozesssteuersystems zu erzeugen. Der beispielhafte Prozess von [Fig. 5](#) beginnt damit, dass der beispielhafte Datensammler **405** über den beispielhaften Zugangsserver ZS Prozesseingaben und -ausgaben erhält (Block **505**) und über den beispielhaften Zugangsserver ZS Nutzereingaben erhält (Block **510**). Der beispielhafte Datensammler **405** speichert die erhaltenen Daten in der beispielhaften Datenbank **410**.

[0054] Der beispielhafte Datenmodifizierer **415** erzeugt aus den vom Datensammler **405** erhaltenen Daten einen anfänglichen zeitindizierten Datensatz (Block **515**). Der Datenmodifizierer **415** skaliert bei Bedarf beliebige der Prozesseingabewerte und/oder ersetzt den Text benannter Mengen durch numerische Werte (Block **520**). Der Datenmodifizierer **415** komprimiert den zeitindizierten Datensatz, um Einträge zu entfernen, die keine geänderten Eingaben darstellen (Block **525**). Der komprimierte zeitindizierte Datensatz wird in der Datenbank **410** gespeichert und die Steuerung verlässt den beispielhaften Prozess von [Fig. 5](#).

[0055] Der beispielhafte Prozess von [Fig. 6](#) kann ausgeführt werden, um ein Prozesssteuersystem zu testen. Der beispielhafte Prozess von [Fig. 6](#) beginnt damit, dass der beispielhafte Testserver TS den Zustand des zu testenden Prozesssteuersystems unter Verwendung zum [sic!] eines Downloads im DeltaV Explorer oder einer Wiederherstellung im DeltaV SimulatePro (Block **605**) initialisiert. Der beispielhafte Wiedergeber **420** führt das zu testende Prozesssteuersystem aus, setzt es in Gang und/oder betreibt es, indem er die im komprimierten zeitindizierten Datensatz enthaltenen Eingabewerte über den Zugangs server ZS zum Prozesssteuersystem sendet (Block **610**).

[0056] Der Datensammler **405** erhält Nutzereingaben, Prozesseingaben und Prozessausgaben, die von den Event-Chronisten EC und den Data Historians DH während der Ausführung von Block **610** erfasst wurden (Block **615**). Der Datensammler **405** kann die Daten zum Beispiel durch Ausführen des beispielhaften Prozesses von [Fig. 5](#) erhalten. Der beispielhafte Vergleicher **425** vergleicht die ursprünglichen und die zusätzlich gesammelten Prozessausgaben, um festzustellen, ob das zu testende Prozesssteuersystem wie vorgesehen gelaufen ist (Block **620**). Dann verlässt die Steuerung den beispielhaften Prozess von [Fig. 6](#).

[0057] In einigen Beispielen kann der beispielhafte Prozess von [Fig. 6](#) mehr als einmal ausgeführt werden. Zum Beispiel kann der Prozess von [Fig. 6](#) ein erstes Mal ausgeführt werden, um den vorgesehenen Betrieb einer zweiten Prozesssteueranlage zu verifizieren, die sich nur insofern von einer ersten Prozesssteueranlage unterscheidet, als das zweite Prozesssteuersystem separat von der Verarbeitungsanlage des Kunden betrieben wird. Der Prozess von [Fig. 6](#) könnte ein zweites Mal ausgeführt werden, um den vorgesehenen Betrieb eines dritten Prozesssteuersystems zu verifizieren, das die zweite Prozesssteuerung nach Anwendung einer Reparatur, Korrektur, eines Upgrades usw. ist.

[0058] [Fig. 7](#) ist ein Blockdiagramm einer beispielhaften Prozessorplattform P100, die zur Implemen-

tierung der Testserver TS, der beispielhaften Bedienstation **115**, des beispielhaften Arbeitsplatzes **120**, der beispielhaften Anwendungsstation **121** und/oder zur Ausführung eines oder aller der hierin beschriebenen beispielhaften maschinenzugänglichen Befehle und/oder Prozesse verwendet und/oder programmiert sein kann. Zur Implementierung der Prozessorplattform P100 können ein oder mehrere Universalprozessoren, Prozessorkerne, Mikrocontroller usw. verwendet werden. Die Prozessorplattform P100 kann zum Beispiel ein Server, ein PC, ein Embedded Controller und/oder jede andere Art von Rechnergerät sein.

[0059] Die Prozessorplattform P100 des vorliegenden Beispiels weist mindestens einen programmierbaren Prozessor P105 auf. Der Prozessor P105 führt codierte Befehle P110 und/oder P112 aus, die im Hauptspeicher des Prozessors P105 (z. B. in einem RAM P115 und/oder einem ROM P120) vorhanden sind. Der Prozessor P105 kann jede beliebige Art einer Verarbeitungseinheit sein, wie etwa ein Prozessorkern, ein Prozessor und/oder ein Mikrocontroller. Der Prozessor P105 kann dafür programmiert sein, die beispielhaften Prozesse von [Fig. 5](#) und [Fig. 6](#) auszuführen. So können die codierten Befehle P110, P112 die beispielhaften Prozesse von [Fig. 5](#) und/oder [Fig. 6](#) darstellen.

[0060] Der Prozessor P105 kommuniziert über einen Bus P125 mit dem Hauptspeicher, der einen ROM P110 und den RAM P115 aufweist. Der RAM P115 kann durch einen synchronen dynamischen Speicher mit wahlfreiem Zugriff (SDRAM), einen dynamischen RAM (DRAM), einen synchronen DRAM von RAM-BUS (RDRAM) und/oder durch jede andere Art von RAM-Gerät implementiert sein. Der ROM P110 kann durch einen Flash-Speicher und/oder durch jede andere gewünschte Art von Speichergerät implementiert sein. Der Zugriff auf den Speicher P115 und den Speicher P120 kann durch einen Speichercontroller gesteuert sein. Der beispielhafte Speicher P115 kann verwendet werden, um zum Beispiel die beispielhafte Datenbank **410** zu speichern.

[0061] Die Prozessorplattform P100 weist einen Schnittstellenschaltkreis P130 auf. Jede beliebige Art von Schnittstellenstandard wie etwa eine externe Speicherschnittstelle, ein serieller Port, eine universelle Eingabe/Ausgabe, eine Ethernet-Schnittstelle, ein universeller serieller Bus (USB) und/oder eine PCI-Express-Schnittstelle usw. kann den Schnittstellenschaltkreis P130 implementieren.

[0062] An den Schnittstellenschaltkreis P130 können ein oder mehrere Eingabegeräte P135 angeschlossen sein. Das Eingabegerät bzw. die Eingabegeräte P135 können zum Beispiel durch eine Tastatur, eine Maus, einen Berührungsbildschirm, ein Tastfeld, eine Rollkugel, Isopoint und/oder ein

Stimmenerkennungssystem implementiert sein. An den Schnittstellenschaltkreis 1020 [sic!] sind auch ein oder mehrere Ausgabegeräte P140 angeschlossen. Die Ausgabegeräte P140 können zum Beispiel durch Anzeigegeräte (z. B. eine Flüssigkristallanzeige, einen Röhren-(CRT-)Monitor, einen Drucker und/oder Lautsprecher) implementiert sein. So kann der Schnittstellenschaltkreis P130 eine Grafiktreiberkarte aufweisen. Das Eingabegerät bzw. die Eingabegeräte P135 und das Ausgabegerät bzw. die Ausgabegeräte P136 können zusätzlich oder alternativ auch zum Implementieren des beispielhaften Datensammlers **405** verwendet werden.

[0063] Der Schnittstellenschaltkreis P130 kann auch ein Kommunikationsgerät wie etwa ein Modem, einen Sende-Empfänger oder eine Netzwerkschnittstellenkarte aufweisen, um den Austausch von Daten mit externen Computern über ein Netzwerk (z. B. das LAN **125**) zu erleichtern.

[0064] In einigen Beispielen weist die Prozessorplattform P100 auch ein oder mehrere Massenspeichergeräte P145 zum Speichern von Software und Daten auf. Beispiele solcher Massenspeichergeräte P145 sind u. a. ein Diskettenlaufwerk, ein Festplattenlaufwerk, ein Festkörperlaufwerk, ein CD-Laufwerk, ein DVD-Laufwerk und/oder jedes andere Festkörper-, Magnet- und/oder optisches Speichergerät. Das Massenspeichergerät P145 kann die beispielhafte Betriebsdatenbank **155** und/oder die beispielhafte Datenbank **410** implementieren.

[0065] Die hierin beschriebenen codierten Befehle können im Massenspeichergerät P145, im RAM P115, im ROM P120 und/oder auf einem mobilen Speichermedium wie etwa einer CD oder einer DVD gespeichert sein.

[0066] Aus dem Vorangegangenen wird man verstehen, dass die offenbarten Verfahren, Vorrichtungen und Fertigungsartikel zum Testen von Prozesssteuersystemen [sic!]. Obschon hierin bestimmte beispielhafte Verfahren, Vorrichtungen und Fertigungsartikel beschrieben wurden, ist der Geltungsbereich des vorliegenden Patents nicht auf diese beschränkt. Im Gegenteil erfasst das vorliegende Patent alle Verfahren, Vorrichtungen und Fertigungsartikel, die nach billigem Ermessen in den Geltungsbereich der Ansprüche des vorliegenden Patents fallen.

Patentansprüche

1. Verfahren, Folgendes umfassend:
Erhalten von Nutzereingaben in ein erstes Prozesssteuersystem,
Erhalten von Prozesseingaben und ersten Prozessausgaben des ersten Prozesssteuersystems,

Liefern der Nutzereingaben und der Prozesseingaben zu einem zweiten Prozesssteuersystem, um das zweite Prozesssteuersystem zu betreiben,
 Erhalten zweiter Prozessausgaben des zweiten Prozesssteuersystems, das mit den Nutzereingaben und den Prozesseingaben betrieben wurde, und
 Vergleichen der ersten und der zweiten Prozessausgaben, um festzustellen, ob das zweite Prozesssteuersystem wie vorgesehen implementiert wird.

2. Verfahren nach Anspruch 1, ferner Folgendes umfassend:

Liefern der Nutzereingaben und der Prozesseingaben zu einem dritten Prozesssteuersystem, um das dritte Prozesssteuersystem zu betreiben,
 Erhalten dritter Prozessausgaben von dem dritten Prozesssteuersystem, das mit den Nutzereingaben und den Prozesseingaben betrieben wurde, und
 Vergleichen der zweiten und der dritten Prozessausgaben, um festzustellen, ob das dritte Prozesssteuersystem wie vorgesehen implementiert wird.

3. Verfahren nach Anspruch 1, wobei das Vergleichen der ersten und der zweiten Prozessausgaben, um festzustellen, ob das zweite Prozesssteuersystem wie vorgesehen implementiert wird, Folgendes umfasst:

Berechnen einer Differenz zwischen einer ersten der ersten Prozessausgaben und einer ersten der zweiten Prozessausgaben und
 Vergleichen der Differenz mit einem Schwellenwert.

4. Verfahren nach Anspruch 3, wobei der Schwellenwert mindestens entweder einen Prozentsatz der ersten der ersten Prozessausgaben und/oder eine Zeitmarkendifferenz darstellt.

5. Verfahren nach Anspruch 1, ferner das Erfassen der Nutzereingaben, der Prozesseingaben und der ersten Prozessausgaben umfassend, während das erste Prozesssteuersystem läuft.

6. Verfahren nach Anspruch 1, wobei das erste Prozesssteuersystem an einer Verarbeitungsanlage implementiert wird und das zweite Prozesssteuersystem in einer Testeinrichtung implementiert wird.

7. Verfahren nach Anspruch 1, wobei das zweite Prozesssteuersystem ein Schulungssystem umfasst.

8. Verfahren nach Anspruch 1, wobei das zweite Prozesssteuersystem das erste Prozesssteuersystem umfasst und das zweite Prozesssteuersystem eine andere Software-Version als das erste Prozesssteuersystem implementiert.

9. Verfahren nach Anspruch 1, ferner Folgendes umfassend:

Kombinieren der Nutzereingaben und der Prozesseingaben zu einem Datensatz, wobei der Datensatz zeitlich indiziert wird, und
 Skalieren einer ersten der Prozesseingaben, um eine Modifizierung der ersten der Prozesseingaben durch das erste Prozesssteuersystem zu kompensieren.

10. Verfahren nach Anspruch 9, ferner das Ersetzen einer zweiten der Prozesseingaben durch einen numerischen Wert umfassend, wenn die zweite der Prozesseingaben Text umfasst.

11. Verfahren nach Anspruch 9, ferner das Komprimieren des Datensatzes umfassend, um ein Element des Datensatzes zu entfernen, das keiner der Nutzereingaben, keiner der Prozesseingaben und keiner der ersten Prozessausgaben entspricht.

12. Vorrichtung, Folgendes umfassend:
 einen Datensammler zum Erhalten von Nutzereingaben, Prozesseingaben und ersten Prozessausgaben des ersten Prozesssteuersystems und zum Erhalten zweiter Prozessausgaben eines zweiten Prozesssteuersystems,
 einen Wiedergeber zum Liefern der Nutzereingaben und der Prozesseingaben zum zweiten Prozesssteuersystem, um das zweite Prozesssteuersystem zu betreiben, und
 einen Vergleicher zum Vergleichen der ersten und der zweiten Prozessausgaben, um festzustellen, ob das zweite Prozesssteuersystem wie vorgesehen gelaufen ist.

13. Vorrichtung nach Anspruch 12, ferner Folgendes umfassend:

einen Event-Chronisten zum Erfassen der Nutzereingaben, während das erste Prozesssteuersystem läuft, und
 einen Data Historian zum Erfassen der Prozesseingaben und der ersten Prozessausgaben, während das erste Prozesssteuersystem läuft.

14. Vorrichtung nach Anspruch 12, wobei das erste Prozesssteuersystem innerhalb einer Verarbeitungsanlage betrieben wird.

15. Vorrichtung nach Anspruch 12, wobei das zweite Prozesssteuersystem das erste Prozesssteuersystem umfasst und das zweite Prozesssteuersystem eine andere Software-Version als das erste Prozesssteuersystem implementiert.

16. Vorrichtung nach Anspruch 12, wobei der Datensammler zum Kombinieren der Nutzereingaben und der Prozesseingaben zu einem Datensatz dient, wobei der Datensatz zeitlich indiziert ist, und die ferner einen Datenmodifizierer zum Skalieren einer ersten der Prozesseingaben umfasst, um eine Modifizierung der ersten der Prozesseingaben durch das erste Prozesssteuersystem zu kompensieren.

17. Vorrichtung nach Anspruch 16, wobei der Datenmodifizierer zum Ersetzen einer zweiten der Prozesseingaben durch einen numerischen Wert dient, wenn die zweite der Prozesseingaben Text umfasst.

18. Physischer Fertigungsartikel, der maschinenlesbare Befehle speichert, welche bei der Ausführung bewirken, dass ein Prozessor mindestens:

Nutzereingaben in ein erstes Prozesssteuersystem erhält,

Prozesseingaben und erste Prozessausgaben des ersten Prozesssteuersystems erhält,

die Nutzereingaben und die Prozesseingaben zu einem zweiten Prozesssteuersystem liefert, um das zweite Prozesssteuersystem zu betreiben,

zweite Prozessausgaben des zweiten Prozesssteuersystems erhält, das mit den Nutzereingaben und den Prozesseingaben betrieben wurde,

die Nutzereingaben und die Prozesseingaben zu einem dritten Prozesssteuersystem liefert, um das dritte Prozesssteuersystem zu betreiben,

dritte Prozessausgaben von dem dritten Prozesssteuersystem erhält, das mit den Nutzereingaben und den Prozesseingaben betrieben wurde, und

die zweiten und die dritten Prozessausgaben vergleicht, um festzustellen, ob das dritte Prozesssteuersystem wie vorgesehen implementiert ist.

19. Physischer Fertigungsartikel nach Anspruch 18, wobei das erste Prozesssteuersystem an einer Verarbeitungsanlage implementiert ist und das zweite und das dritte Prozesssteuersystem in einer Testeinrichtung implementiert sind.

20. Physischer Fertigungsartikel nach Anspruch 18, wobei das zweite Prozesssteuersystem das erste Prozesssteuersystem umfasst und das dritte Prozesssteuersystem eine andere Software-Version als das zweite Prozesssteuersystem implementiert.

Es folgen 6 Blatt Zeichnungen

Anhängende Zeichnungen

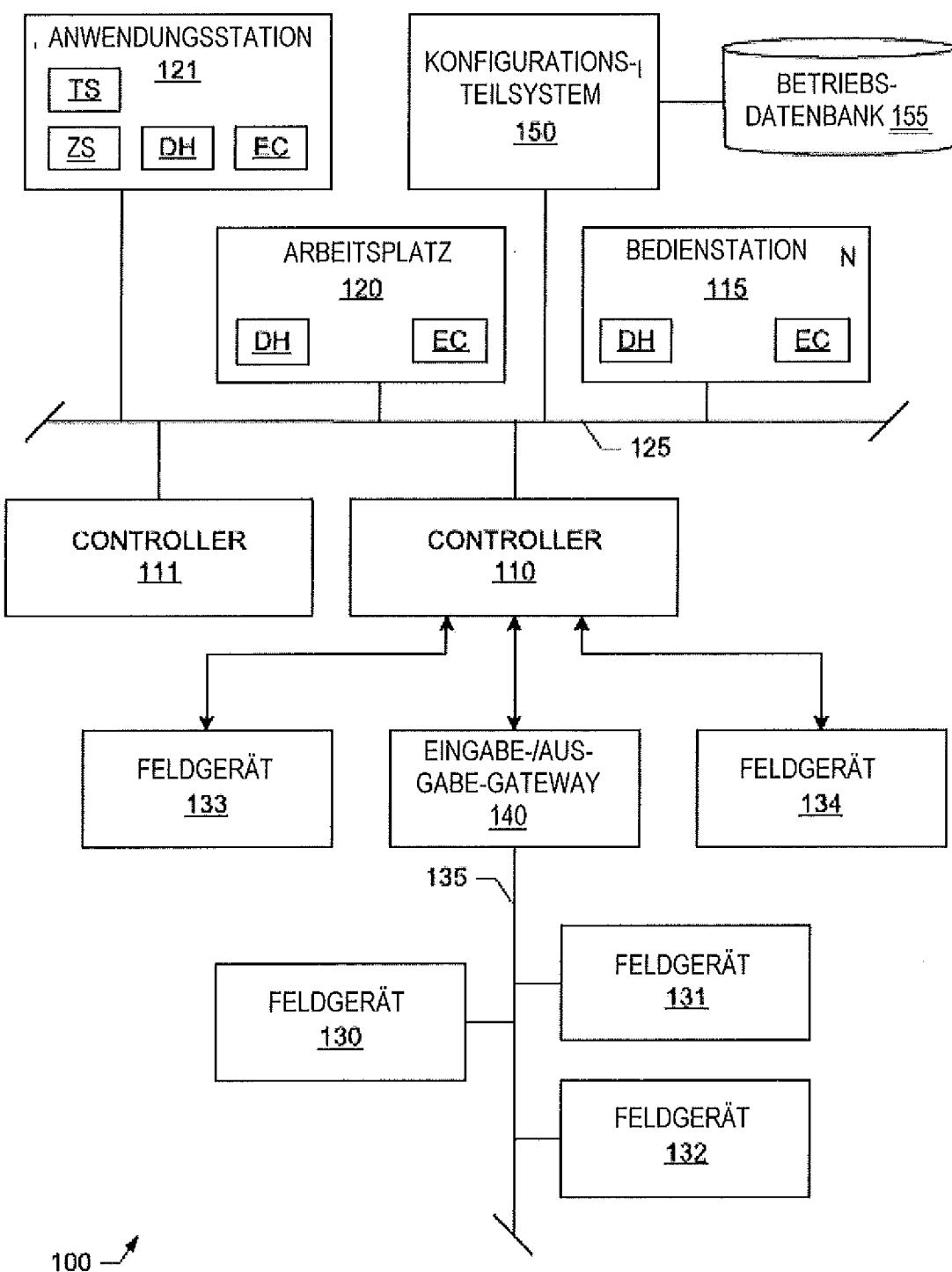


FIG. 1

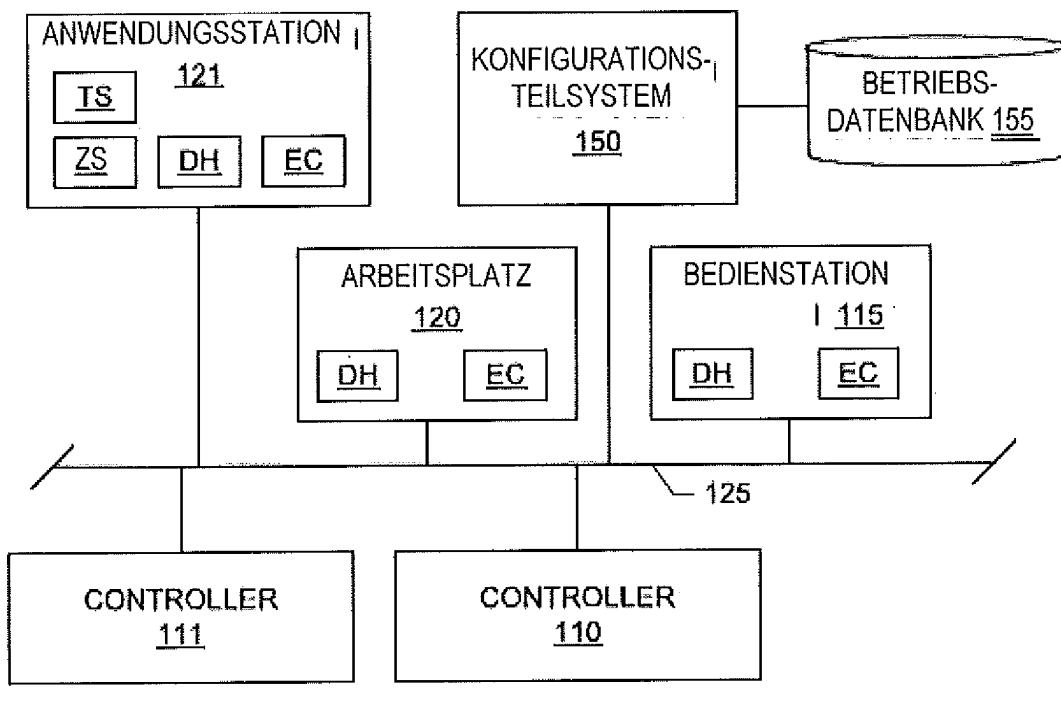


FIG. 2

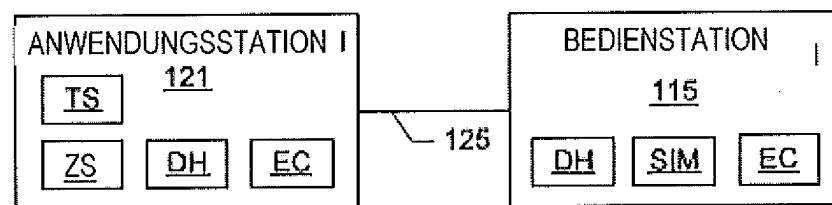


FIG. 3

→ 300

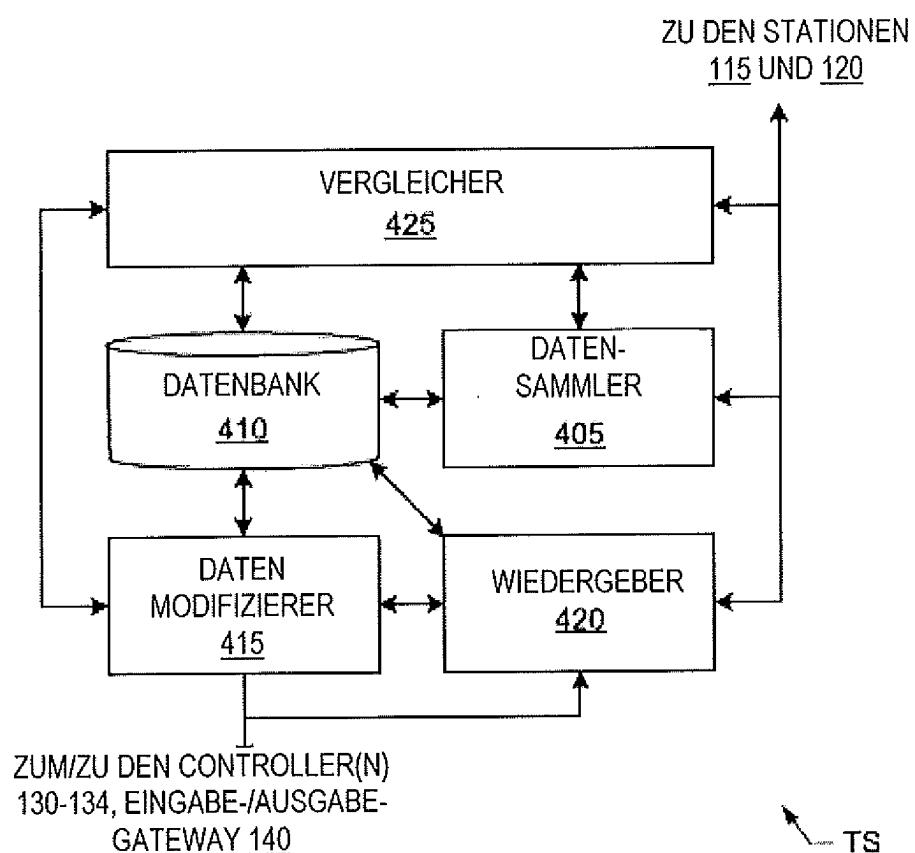


FIG. 4

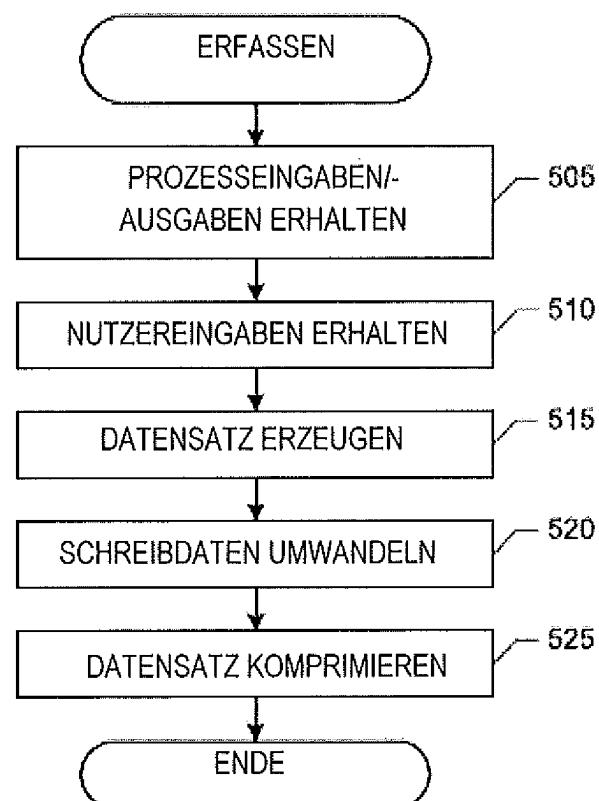


FIG. 5

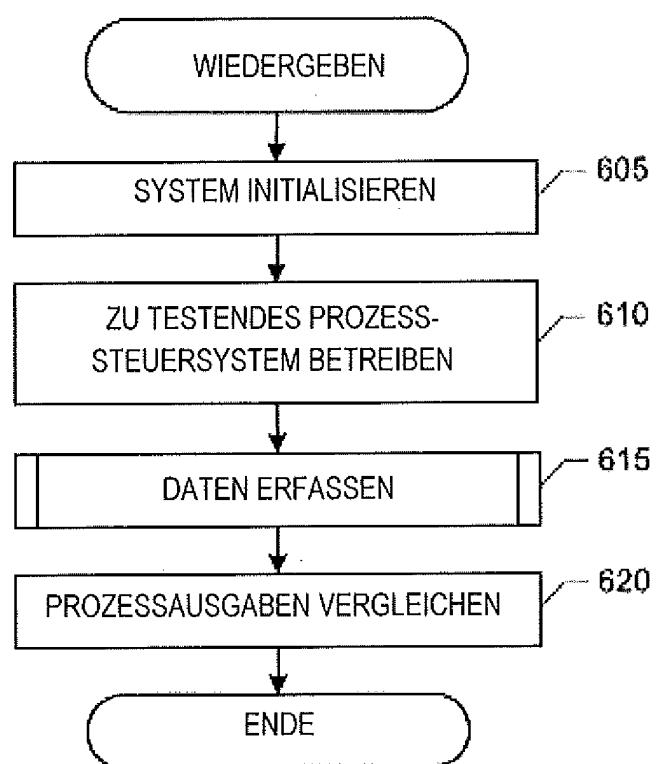


FIG. 6

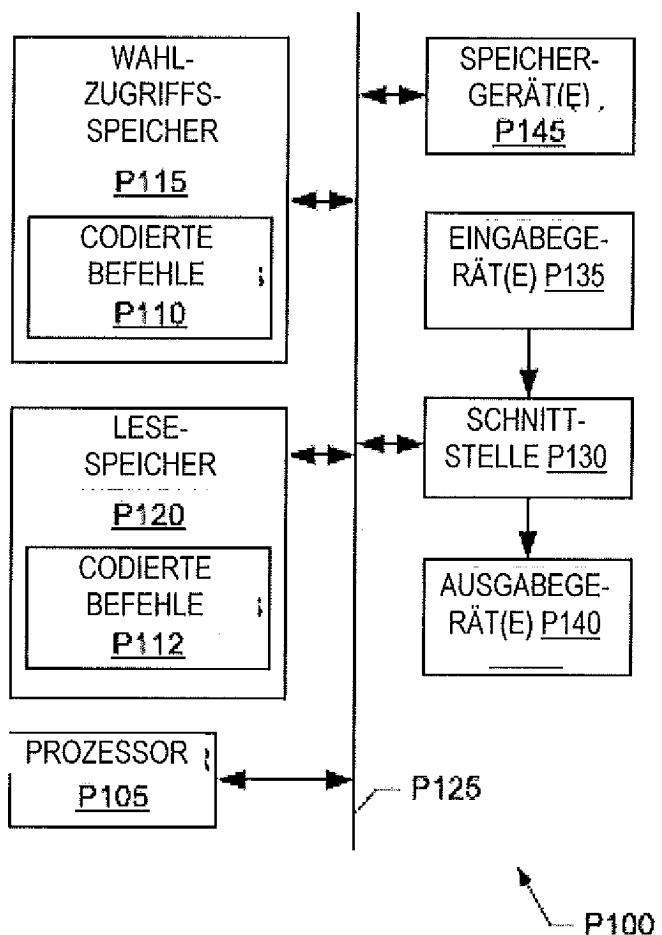


FIG. 7