



(19)  
Bundesrepublik Deutschland  
Deutsches Patent- und Markenamt

(10) **DE 697 32 488 T2 2006.03.30**

(12) **Übersetzung der europäischen Patentschrift**

(97) **EP 0 931 284 B1**

(51) Int Cl.<sup>8</sup>: **G05B 19/418 (2006.01)**

(21) Deutsches Aktenzeichen: **697 32 488.5**

(86) PCT-Aktenzeichen: **PCT/US97/17983**

(96) Europäisches Aktenzeichen: **97 945 512.8**

(87) PCT-Veröffentlichungs-Nr.: **WO 98/014853**

(86) PCT-Anmeldetag: **03.10.1997**

(87) Veröffentlichungstag  
der PCT-Anmeldung: **09.04.1998**

(97) Erstveröffentlichung durch das EPA: **28.07.1999**

(97) Veröffentlichungstag  
der Patenterteilung beim EPA: **09.02.2005**

(47) Veröffentlichungstag im Patentblatt: **30.03.2006**

(30) Unionspriorität:  
**726266 04.10.1996 US**

(84) Benannte Vertragsstaaten:  
**DE, FI, FR, GB, SE**

(73) Patentinhaber:  
**Fisher Controls International LLC, St. Louis, Mo.,  
US**

(72) Erfinder:  
**BURNS, A., Harry, Marshalltown, US; LARSON, H.,  
Brent, Marshalltown, US; BROWN, K., Larry,  
Marshalltown, US**

(74) Vertreter:  
**Meissner, Bolte & Partner GbR, 80538 München**

(54) Bezeichnung: **PROZESSSTEUERUNGSNETZWERK MIT REDUNDANTEN FELDDGERÄTEN UND BUSSEN**

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

**Beschreibung**

## VERWANDTE ANMELDUNG

**[0001]** Dies ist eine Teilfortführung der US-Patentanmeldung Ser.-Nr. 08/726 266, angemeldet am 4. Oktober 1996.

## GEBIET DER ERFINDUNG

**[0002]** Die vorliegende Erfindung betrifft allgemein Prozeßsteuerungsnetze und speziell ein Prozeßsteuerungsnetz, das Prozeßsteuerungsfunktionen auf verteilte Weise implementiert unter Verwendung von redundanten Funktionselementen wie etwa Feldgeräten und Kommunikationsbussen.

## BESCHREIBUNG DES STANDS DER TECHNIK

**[0003]** Großprozesse wie chemische, Erdöl- und andere Herstellungs- und Raffinationsprozesse weisen zahlreiche Feldgeräte auf, die an verschiedenen Stellen angeordnet sind, um Parameter eines Prozesses zu messen und zu steuern und dadurch die Steuerung des Prozesses auszuführen. Diese Feldgeräte können beispielsweise Sensoren wie etwa Temperatur-, Druck- und Durchflußratensensoren sowie Steuerelemente wie Ventile und Schalter sein. Historisch hat die Prozeßsteuerungsindustrie manuelle Vorgänge angewandt wie etwa das manuelle Ablesen von Pegel- und Druckmessern, das Drehen von Ventilrädern usw., um die Messung durchzuführen und Feldgeräte innerhalb eines Prozesses zu steuern. Mit dem Beginn des 20. Jahrhunderts begann die Prozeßsteuerungsindustrie damit, die lokale pneumatische Steuerung anzuwenden, wobei lokale pneumatische Steuerelemente, Meßwertgeber und Ventilpositionierer an verschiedenen Stellen innerhalb einer Prozeßanlage angeordnet wurden, um die Steuerung bestimmter Anlagenlokationen auszuführen. Mit dem Aufkommen des mikroprozessor-basierenden verteilten Steuersystems (DCS) in den siebziger Jahren des letzten Jahrhunderts wurde die verteilte elektronische Prozeßsteuerung in der Prozeßsteuerungsindustrie vorherrschend.

**[0004]** Wie bekannt ist, weist ein DCS einen Analog- oder einen Digitalrechner wie etwa eine programmierbare logische Steuereinrichtung auf, die mit zahlreichen elektronischen Überwachungs- und Steuereinrichtungen wie elektronischen Sensoren, Meßwertgebern, Strom-Druck-Wandlern, Ventilpositionierern usw., die innerhalb eines Prozesses angeordnet sind, verbunden ist. Der DCS-Rechner speichert und implementiert ein zentralisiertes und häufig komplexes Steuerungsschema, um die Messung und Steuerung von Geräten innerhalb des Prozesses auszuführen und dadurch Prozeßparameter in Abhängigkeit von einem Gesamtsteuerungsschema zu steuern. Gewöhnlich ist jedoch das von einem DCS imp-

lementierte Steuerungsschema Eigentum des Herstellers der DCS-Steuereinrichtung, was wiederum eine Erweiterung, ein Aufrüsten, Umprogrammieren und den Service schwierig und teuer macht, weil der DCS-Lieferant auf integrale Weise involviert werden muß, damit diese Aktivitäten ausgeführt werden können. Außerdem können die Einrichtungen, die innerhalb eines bestimmten DCS verwendet oder in dasselbe eingekoppelt sein können, möglicherweise aufgrund der herstellerspezifischen Beschaffenheit der DCS-Steuereinrichtung und durch die Tatsache begrenzt sein, daß ein Lieferant der DCS-Steuereinrichtung bestimmte Einrichtungen oder Funktionen von Einrichtungen, die von anderen Herstellern stammen, eventuell nicht unterstützt.

**[0005]** Zur Lösung einiger der Probleme, die bei der Verwendung von herstellerspezifischen DCS auftreten, hat die Prozeßsteuerungsindustrie eine Reihe von offenen Standard-Kommunikationsprotokollen entwickelt, die beispielsweise die HART<sup>®</sup>-, PROFIBUS<sup>®</sup>-, WORLDFIP<sup>®</sup>-, Device-Net<sup>®</sup>- und CAN-Protokolle umfassen, die es möglich machen, daß von verschiedenen Herstellern stammende Feldgeräte innerhalb desselben Prozeßsteuerungsnetzes gemeinsam verwendet werden können. Tatsächlich kann jedes Feldgerät, das einem dieser Protokolle entspricht, innerhalb eines Prozesses verwendet werden, um mit einer DCS-Steuereinrichtung oder einer anderen das Protokoll unterstützenden Steuereinrichtung zu kommunizieren und davon gesteuert zu werden, auch wenn dieses Feldgerät von einem anderen Hersteller als dem Hersteller der DCS-Steuereinrichtung stammt.

**[0006]** Ferner gibt es heute eine Bewegung innerhalb der Prozeßsteuerungsindustrie in Richtung einer Dezentralisierung der Prozeßsteuerung und dadurch einer Vereinfachung von DCS-Steuereinrichtungen oder der Eliminierung der Notwendigkeit für DCS-Steuereinrichtungen in größerem Umfang. Eine dezentrale Steuerung wird erhalten, indem man im Feld angebrachte Prozeßsteuerungseinrichtungen wie etwa Ventilpositionierer, Meßwertgeber usw. eine oder mehrere Prozeßsteuerfunktionen ausführen läßt und dann Daten über eine Busstruktur übermittelt zur Nutzung durch andere Prozeßsteuerungseinrichtungen bei der Durchführung von anderen Steuerfunktionen. Zur Implementierung dieser Steuerfunktionen weist jede Prozeßsteuerungseinrichtung einen Mikroprozessor auf, der die Fähigkeit hat, eine Steuerfunktion auszuführen, und auch die Fähigkeit hat, mit anderen Prozeßsteuerungseinrichtungen unter Anwendung eines offenen und standardisierten Kommunikationsprotokolls zu kommunizieren. Auf diese Weise können Feldgeräte, die von verschiedenen Herstellern stammen, in einem Prozeßsteuerungsnetz miteinander gekoppelt sein, um miteinander zu kommunizieren und eine oder mehrere Prozeßsteuerfunktionen auszuführen unter Bildung einer

Steuerschleife, ohne daß eine DCS-Steuereinrichtung zwischengeschaltet ist. Das volldigitale Zweidraht-Busprotokoll, das heute von der Fieldbus Foundation verbreitet wird und als das FOUNDATION™ Fieldbus-Protokoll (nachstehend "Fieldbus"-Protokoll) bekannt ist, ist ein offenes Kommunikationsprotokoll, das es Einrichtungen, die von verschiedenen Herstellern stammen, möglich macht, über einen Standardbus zusammenzuwirken und miteinander zu kommunizieren, um eine dezentrale Steuerung innerhalb eines Prozesses auszuführen.

**[0007]** In allen komplexen Prozeßsteuerungssystemen können Probleme sowohl im Prozeßsteuerungsnetz als auch in den Prozeßsteuerungselementen auftreten. Zur Lösung dieser Probleme ist die Anwendung von Redundanz in Kommunikationssystemen in den folgenden zitierten Dokumenten erörtert worden: US 4 627 045 und US 5 148 433. Diese Dokumente beschreiben die Verwendung redundanter Prozessoren und redundanter Kommunikationskanäle. Insbesondere beschreibt US 4 627 045 ein System, in dem eine Vielzahl von Stationen mit redundanten Kommunikationskanälen gekoppelt sind und jede Station bestimmt, auf welchem Kommunikationskanal Daten zu übertragen sind. US 5 148 433 beschreibt ein redundantes Prozessorsystem, das mit einem Kommunikationsnetz gekoppelt ist, wobei das redundante Prozessorsystem einen aktiven Prozessor und einen Sicherungsprozessor aufweist. Der aktive Prozessor empfängt und überträgt Daten über das Kommunikationsnetz für das redundante Prozessorsystem. Wenn der Sicherungsprozessor feststellt, daß der aktive Prozessor ausgefallen ist, nimmt der Sicherungsprozessor den aktiven Status an.

**[0008]** Ungeachtet des verwendeten Kommunikationsprotokolls werden Prozeßsteuerungselemente wie etwa Fluidsteuerventile gewöhnlich in rauen Prozeßsteuerungsumgebungen eingesetzt, in denen Temperatur- und Druckbereiche stark veränderlich sind. Anwendungsgebiete von Fluidsteuerventilen, bei denen raue Umgebungen üblich sind, umfassen Anwendungsgebiete in Öl- und Gasleitungen, in Kernkraftwerken und verschiedenen Prozeßsteuerungsanwendungen. In diesen Umgebungen sind erhebliche Wartungsarbeiten üblich, was periodische präventive Wartung, Wartung infolge eines Ventilausfalls und Tests umfaßt, um festzustellen, daß die Ventile ordnungsgemäß arbeiten.

**[0009]** In diesen rauen Umgebungen ermüden Steuerelemente oder fallen aus und müssen hin und wieder ersetzt werden. Sowohl der Ausfall eines Steuerelements als auch der Austausch eines Steuerelements erfordert typischerweise die Abschaltung des Prozeßsteuerungssystems, was außerordentlich teuer und zeitaufwendig ist infolge der langen Zeitintervalle, die erforderlich sind, um nach der Abschaltung das Prozeßsteuerungssystem wieder in einen

stabilen Zustand zu bringen.

**[0010]** Es ist daher erwünscht, eine Vorrichtung und ein Betriebsverfahren anzugeben, das es einem Prozeßsteuerungsnetz, das beispielsweise ein schleifengespeistes, digitales Zweidraht-Zweiwege-Kommunikationsprotokoll oder irgendein anderes verteiltes Prozeßfunktionsprotokoll verwendet, erlaubt, ungeachtet des Ausfalls oder der Auswechslung von Funktionselementen in dem Netz funktionsfähig zu bleiben.

#### ZUSAMMENFASSUNG DER ERFINDUNG

**[0011]** Gemäß der vorliegenden Erfindung werden Funktionselemente innerhalb eines Prozeßsteuerungssystems wie etwa einer digitalen schleifengespeisten, Zweidraht-Zweiwege-Kommunikationsumgebung unter Verwendung selektiver redundanter Verbindungen und selektiver redundanter Funktionselemente miteinander verbunden. Die redundanten Funktionselemente und redundanten Verbindungen sorgen für einen reibungslosen Übergang des Betriebs von einem primären Prozeßschleifenelement zu einem sekundären Prozeßschleifenelement im Fall eines Ausfalls des primären Prozeßschleifenelements.

**[0012]** Gemäß einem Aspekt der vorliegenden Erfindung wird die Redundanz selektiv implementiert unter Verwendung von zwei Sets von Kommunikationsmedien, die ein redundantes Buspaar aufweisen, das einen primären Bus und einen redundanten Bus hat. Gemäß einem anderen Aspekt der vorliegenden Erfindung wird die Redundanz selektiv implementiert unter Verwendung eines einzigen Sets von Kommunikationsmedien wie etwa eines einzigen Kommunikationsbusses, jedoch unter Implementierung von redundanten Einrichtungen wie etwa Feldgeräten, so daß bei Ausfall eines Geräts oder eines anderen funktionalen Elements wie etwa eines Funktionsblocks die Wiederherstellung erreicht wird, jedoch nicht für den Ausfall des Kommunikationsmediums. Bei einer Ausführungsform steuert eine Schleifensteuereinheit wie etwa eine Steuereinheit eines digitalen Steuersystems bzw. DCS oder ein Feldgerät die Redundanzoperation einer einzigen Kommunikationsschleife, in der sich redundante funktionale Elemente befinden. Bei dieser Ausführungsform ist die Schleifensteuereinheit mit einem einzigen Kommunikationsbus verbunden, und der einzige Kommunikationsbus ist mit einem redundanten Paar von funktionalen Elementen wie etwa Geräten verbunden. Ausgewählte funktionale Elemente wie Steuerlogik detektieren einen Ausfallstatus und übermitteln diesen Status an eine Steuereinheit, oder die Steuereinheit detektiert eine Unterbrechung der Kommunikationen von einem ausgefallenen von den redundanten funktionalen Elementen und rekonfiguriert automatisch die Kommunikationsschleife, um dadurch den Kom-

munikationsstatus wieder herzustellen.

**[0013]** Gemäß einem weiteren Aspekt der vorliegenden Erfindung wird die Redundanz selektiv implementiert unter Verwendung eines redundanten Sets von Kommunikationsmedien zusätzlich zu der Verwendung von anderen redundanten funktionalen Elementen wie etwa Geräten, so daß die Wiederherstellung sowohl für ein ausgefallenes Gerät als auch ein ausgefallenes Kommunikationsmedium erreicht wird. Gemäß der vorliegenden Erfindung steuert eine Schleifensteuereinheit wie etwa eine Steuereinheit eines digitalen Steuersystems bzw. DCS oder eines Feldgeräts die Redundanzoperation eines redundanten Paares von Kommunikationsschleifen, die redundante Busse haben, die mit redundanten Geräten verbunden sind. Die Schleifensteuereinheit ist sowohl mit einem primären Bus als auch einem redundanten Bus des redundanten Paares von Kommunikationsschleifen verbunden, und die redundanten Geräte sind mit den redundanten Bussen verbunden, so daß ein primäres Gerät mit der primären Schleife verbunden ist und ein redundantes Gerät mit der redundanten Schleife verbunden ist. Ausgewählte funktionale Elemente detektieren und übermitteln einen Ausfallstatus an die Schleifensteuereinheit, oder die Schleifensteuereinheit detektiert eine Unterbrechung von Kommunikationen von einem ausgefallenen funktionalen Element. Bei einem Ausfall, wenn beispielsweise die Steuereinheit oder Steuerlogik ein ausgefallenes funktionales Element (entweder einen Bus oder ein Gerät) detektiert oder die Schleifensteuereinheit einen Abbruch von Kommunikationen von einem Element detektiert, rekonfiguriert die Schleifensteuereinheit automatisch das redundante Paar von Kommunikationsschleifen, um den Kommunikationsstatus wieder herzustellen.

**[0014]** Gemäß noch einem weiteren Aspekt der vorliegenden Erfindung wird die Redundanz selektiv implementiert unter Verwendung eines redundanten Sets von Kommunikationsmedien, die mit einem einzelnen Gerät verbunden sind, so daß eine Wiederherstellung für ein ausgefallenes Kommunikationsmedium, jedoch nicht für ein ausgefallenes Gerät erzielt wird. Gemäß der vorliegenden Erfindung steuert eine Schleifensteuereinheit wie etwa eine DCS-Steuereinheit oder ein Feldgerät die Redundanzoperation des redundanten Kommunikationsmediums. Die Schleifensteuereinheit ist sowohl mit einem primären als auch einem redundanten Bus des redundanten Paares von Kommunikationsmedien verbunden, während eine Vielzahl von anderen funktionalen Elementen wie etwa Geräte mit dem redundanten Paar von Kommunikationsmedien verbunden sind. Ausgewählte funktionale Elemente detektieren einen schlechten Kommunikationsstatus, und die Schleifensteuerung detektiert einen Stopp von Kommunikationen. Bei dieser Konfiguration rekonfiguriert die Schleifensteuereinheit automatisch das redundante

Paar von Kommunikationsmedien, wenn ein funktionales Element einen schlechten Kommunikationsstatus detektiert oder die Schleifensteuereinheit einen Stopp von Kommunikationen von einem Element detektiert, um dadurch den Kommunikationsstatus wieder herzustellen.

**[0015]** Durch das beschriebene Prozeßsteuerungssystem und das Operationsverfahren werden viele Vorteile erzielt. Beispielsweise ist es vorteilhaft, daß die Abschaltung einer Prozeßsteuerleitung vermieden wird, wenn bei einem Prozeßgerät oder einem Kommunikationsbus Probleme auftreten. Es ist außerdem vorteilhaft, daß die Selbstdiagnose-Funktionalität von funktionalen Elementen innerhalb des Prozeßsteuerungssystems genutzt wird, um automatisch ausgefallene Elemente zu deaktivieren und den Ersatz von funktionalen Elementen zu aktivieren. Ebenso ist es vorteilhaft, daß das Zweiwege-Kommunikationsprotokoll des Prozeßsteuerungssystems genutzt wird, so daß redundante funktionale Elemente bei Ausfall eines primären funktionalen Elements automatisch aktiviert werden.

#### KURZE BESCHREIBUNG DER ZEICHNUNGEN

**[0016]** [Fig. 1](#) ist ein schematisches Blockbild eines beispielhaften Prozeßsteuerungsnetzes unter Verwendung des Fieldbus-Protokolls;

**[0017]** [Fig. 2](#) ist ein schematisches Blockbild von drei Fieldbus-Geräten, in denen Funktionsblöcke vorgesehen sind;

**[0018]** [Fig. 3](#) ist ein schematisches Blockbild, das die Funktionsblöcke in einigen der Geräte des Prozeßsteuerungsnetzes von [Fig. 1](#) veranschaulicht;

**[0019]** [Fig. 4](#) ist ein Steuerschleifenschema für eine Prozeßsteuerschleife in dem Prozeßsteuerungsnetz von [Fig. 1](#);

**[0020]** [Fig. 5](#) ist ein Zeitdiagramm für einen Makrozyklus eines Segments des Busses des Prozeßsteuerungsnetzes von [Fig. 1](#);

**[0021]** [Fig. 6](#) ist ein schematisches Blockbild, das ein Steuersystemnetz zeigt, in dem die Redundanz selektiv implementiert ist unter Verwendung eines einzigen Satzes von Kommunikationsmedien in Kombination mit redundanten Einrichtungen;

**[0022]** [Fig. 7](#) ist ein schematisches Blockbild, das ein Steuersystemnetz zeigt, in dem die Redundanz selektiv implementiert ist unter Verwendung eines redundanten Kommunikationsmediums in Kombination mit redundanten Einrichtungen;

**[0023]** [Fig. 8](#) ist ein schematisches Blockbild, das ein Steuersystemnetz zeigt, in dem die Redundanz

selektiv implementiert ist unter Verwendung eines redundanten Kommunikationsmediums in Kombination mit einer einzigen Einrichtung;

**[0024]** [Fig. 9](#) ist ein schematisches Blockbild, das ein Steuersystemnetz zeigt, das zwei funktionale Elemente hat, die in eine einzelne Zweidrahtschleife eingekoppelt sind;

**[0025]** [Fig. 10](#) ist ein schematisches Blockbild, das ein Steuersystemnetz zeigt, das zwei Meßwertgeber hat, die in eine einzelne Zweidrahtschleife eingekoppelt sind;

**[0026]** [Fig. 11](#) ist ein schematisches Blockbild, das ein Steuersystemnetz zeigt, das eine redundante Funktionsblockkonfiguration hat;

**[0027]** [Fig. 12](#) ist ein schematisches Blockbild, das ein Steuersystemnetz zeigt, das die Feldgerätredundanz gemäß der vorliegenden Erfindung implementiert;

**[0028]** [Fig. 13](#) ist ein schematisches Blockbild, das ein digitales Feldgerät zeigt, das einen digital kommunizierenden, schleifengespeisten Zweidraht-Zweige-Positionierer zur Verwendung in einem Prozeßsteuerungsnetz gemäß der vorliegenden Erfindung hat; und

**[0029]** [Fig. 14](#) ist ein Blockbild, das eine geeignete Feldgerätsteuereinheit zur Verwendung bei der Steuerung des digitalen Feldgeräts von [Fig. 13](#) zeigt.

#### BESCHREIBUNG DER BEVORZUGTEN AUSFÜHRUNGSFORMEN

**[0030]** Das Prozeßsteuerungsnetz gemäß der Erfindung, das redundante Feldgeräte und Busse aufweist, wird zwar im einzelnen als ein Prozeßsteuerungsnetz beschrieben, das Prozeßsteuerfunktionen auf eine dezentrale oder verteilte Weise unter Verwendung einer Gruppe von Fieldbus-Geräten implementiert, es ist jedoch zu beachten, daß das Prozeßsteuerungsnetz, das redundante Merkmale der vorliegenden Erfindung hat, ein Prozeßsteuerungsnetz sein kann, das verteilte Steuerfunktionen unter Verwendung anderer Arten von Feldgeräten und Kommunikationsprotokollen ausführen kann, was Protokolle einschließt, die auf anderen als Zweidrahtbussen und auf Protokollen basieren, die analoge und digitale Kommunikationen unterstützen. Beispielsweise kann das Prozeßsteuerungsnetz, das redundante Merkmale der vorliegenden Erfindung hat, jedes Prozeßsteuerungsnetz sein, das verteilte Steuerfunktionen ausführt, auch wenn dieses Prozeßsteuerungsnetz das HART-, PROFIBUS-Kommunikationsprotokoll usw. oder irgendwelche anderen Kommunikationsprotokolle verwendet, die es heute gibt bzw. die eventuell noch entwickelt werden.

**[0031]** Bevor die Einzelheiten der Wartungs-Schnittstelleneinrichtung der vorliegenden Erfindung erläutert werden, folgt eine allgemeine Beschreibung des Fieldbus-Protokolls, der nach diesem Protokoll konfigurierten Feldgeräte und der Art und Weise, wie die Kommunikation in einem Prozeßsteuerungsnetz stattfindet, welches das Fieldbus-Protokoll verwendet. Es versteht sich aber, daß das Fieldbus-Protokoll zwar ein relativ neues, volldigitales Kommunikationsprotokoll ist, das zum Gebrauch in Prozeßsteuerungsnetzen entwickelt wurde, dieses Protokoll im Stand der Technik bekannt ist und in zahlreichen Artikeln, Broschüren und Beschreibungen veröffentlicht, verteilt und u. a. zu erhalten ist von der Fieldbus Foundation, einer gemeinnützigen Organisation mit dem Hauptsitz in Austin, Texas. Insbesondere sind das Fieldbus-Protokoll und die Art der Kommunikation mit dieses Protokoll verwendenden Geräten und das Speichern von Daten in diesen im einzelnen in den folgenden Handbüchern beschrieben: Communications Technical Specification und User Layer Technical Specification von der Fieldbus Foundation, die hier summarisch eingeführt werden.

**[0032]** Das Fieldbus-Protokoll ist ein volldigitales, serielles, doppeltgerichtetes Kommunikationsprotokoll, das eine standardisierte physische Schnittstelle für eine Zweidrahtschleife oder einen solchen Bus bereitstellt, der "Feld"-Geräte wie etwa Sensoren, Betätiger, Steuerelemente, Ventile usw. miteinander verbindet, die in einer Instrumenten- oder Prozeßsteuerungsumgebung wie beispielsweise einer Fabrik oder einer Anlage angeordnet sind. Tatsächlich bietet das Fieldbus-Protokoll ein LAN für Feldinstrumente (Feldgeräte) innerhalb eines Prozesses, was es diesen Feldgeräten ermöglicht, Steuerfunktionen an Stellen auszuführen, die innerhalb einer Prozeßanlage verteilt sind, und miteinander zu kommunizieren und nach Ausführung dieser Steuerfunktionen eine Gesamtsteuerungsstrategie zu implementieren. Da es das Fieldbus-Protokoll ermöglicht, daß Steuerfunktionen innerhalb eines Prozeßsteuerungsnetzes verteilt werden, vermindert es die Arbeitslast der zentralen Prozeßsteuereinheit, die typischerweise zu einem DCS gehört, oder beseitigt sogar die Notwendigkeit für eine solche zentrale Prozeßsteuereinheit.

**[0033]** Gemäß [Fig. 1](#) kann ein Prozeßsteuerungsnetz **10**, welches das Fieldbus-Protokoll verwendet, einen Hauptrechner **12** aufweisen, der über eine zweidrahtige Fieldbus-Schleife oder einen solchen Bus **34** mit einer Reihe weiterer Einrichtungen verbunden ist wie etwa mit einer Programmlogiksteuereinheit bzw. PLC **13**, einer Reihe von Steuereinheiten **14**, einer weiteren Hauptrechnereinrichtung **15** und einer Gruppe von Feldgeräten **16, 18, 20, 22, 24, 26, 28, 30** und **32**. Der Bus **34** weist verschiedene Abschnitte oder Segmente **34a, 34b** und **34c** auf, die durch Brückeneinrichtungen **30** und **32** voneinander getrennt sind. Jeder Abschnitt **34a, 34b** und **34c** ver-

bindet eine Untergruppe der mit dem Bus **34** verbundenen Einrichtungen, um die Kommunikation zwischen den Einrichtungen auf eine nachstehend beschriebene Weise zu ermöglichen. Natürlich ist das Netzwerk von [Fig. 1](#) nur beispielhaft, und es gibt viele andere Möglichkeiten, wie ein Prozeßsteuerungsnetz unter Anwendung des Fieldbus-Protokolls konfiguriert sein kann. Typischerweise befindet sich eine Konfiguratoreinheit in einer der Einrichtungen wie etwa dem Hauptrechner **12** und ist dafür zuständig, jedes der Geräte einzurichten oder zu konfigurieren (diese sind "intelligente" Geräte, da sie jeweils einen Mikroprozessor aufweisen, der Kommunikations- und in manchen Fällen auch Steuerfunktionen ausführen kann) sowie zu erkennen, wenn neue Feldgeräte in den Bus **34** eingekoppelt werden, wenn Feldgeräte von dem Bus **34** getrennt werden, Daten zu erkennen, die von den Feldgeräten **16** bis **32** erzeugt werden, und einen Dialog mit einem oder mehreren Anwenderterminals zu führen, die sich in dem Hauptrechner **12** oder in irgendeiner anderen Einrichtung befinden können, die auf irgendeine Weise mit dem Hauptrechner **12** verbunden ist.

**[0034]** Der Bus **34** unterstützt oder ermöglicht eine rein digitale Zweiweg-Kommunikation und kann außerdem ein Energiesignal an einige oder alle Einrichtungen liefern, die daran angeschlossen sind, etwa an die Feldgeräte **16** bis **32**. Alternativ können einige oder alle Geräte **12** bis **32** ihre eigene Energieversorgung haben oder über gesonderte Drähte (nicht gezeigt) an externe Energieversorgungen angeschlossen sein. Die Geräte **12** bis **32** sind in [Fig. 1](#) als an den Bus **34** in einer Verbindung vom Standardbustyp angeschlossen gezeigt, wobei eine Vielzahl von Geräten mit demselben Drahtpaar verbunden sind, das die Bussegmente **34a**, **34b** und **34c** bildet; das Fieldbus-Protokoll erlaubt aber auch andere Geräte-/Draht-Topologien wie etwa Punkt-zu-Punkt-Verbindungen, wobei jedes Gerät mit einer Steuereinheit oder einem Hauptrechner über ein separates Zweidrahtpaar verbunden ist (ähnlich typischen analogen 4-20-mA-DCS-Systemen), und Baum- oder Stichverbindungen, wobei jedes Gerät mit einem gemeinsamen Punkt in einem Zweidrahtbus verbunden ist, der beispielsweise ein Verteilerkasten oder ein Abschlußbereich in einem der Feldgeräte innerhalb eines Prozeßsteuerungsnetzes sein kann.

**[0035]** Daten können in Abhängigkeit von dem Fieldbus-Protokoll über die verschiedenen Bussegmente **34a**, **34b** und **34c** mit den gleichen oder verschiedenen Übertragungsbauraten oder Geschwindigkeiten gesendet werden. Beispielsweise stellt das Fieldbus-Protokoll eine Übertragungsrate (H1) von 31,25 kbit/s, die in der Zeichnung von den Bussegmenten **34b** und **34c** in [Fig. 1](#) verwendet wird, und eine Übertragungsrate (H2) von 1,0 Mbit/s und/oder von 2,5 Mbit/s bereit, die typischerweise für die fortgeschrittene Prozeßsteuerung, Ferneingabe/-ausga-

be und für Hochgeschwindigkeits-Anwendungen bei der Fabrikautomatisierung angewandt und in [Fig. 1](#) von dem Bussegment **34a** verwendet wird. Ebenso können Daten über die Bussegmente **34a**, **34b** und **34c** gemäß dem Fieldbus-Protokoll unter Verwendung der Spannungssignalgabe oder Stromsignalgabe gesendet werden. Selbstverständlich ist die maximale Länge jedes Segments des Busses **34** nicht streng begrenzt, sondern wird statt dessen durch die Übertragungsrate, den Kabeltyp, den Drahtdurchmesser, die Busleistungsoption usw. des jeweiligen Abschnitts bestimmt.

**[0036]** Das Fieldbus-Protokoll klassifiziert die Einrichtungen, die mit dem Bus **34** verbunden werden können, in drei Hauptkategorien, und zwar Grundeinrichtungen, Link-Master bzw. LAS-Einrichtungen (LAS = Link Active Scheduler = link-aktiver Verteiler) und Brückeneinrichtungen. Grundeinrichtungen (wie die Einrichtungen **18**, **20**, **24** und **28** von [Fig. 1](#)) können kommunizieren, d. h. sie können Kommunikationssignale auf dem Bus **34** senden und von diesem empfangen, können jedoch nicht die Reihenfolge oder den Zeitpunkt der Kommunikation, die auf dem Bus **34** stattfindet, steuern. Link-Master-Einrichtungen (wie die Einrichtungen **16**, **22** und **26** sowie der Hauptrechner **12** von [Fig. 1](#)) sind Einrichtungen, die über den Bus **34** kommunizieren und imstande sind, den Fluß und den zeitlichen Ablauf von Kommunikationssignalen auf dem Bus **34** zu steuern. Brückeneinrichtungen (wie die Einrichtungen **30** und **32** von [Fig. 1](#)) sind Einrichtungen, die ausgebildet sind, um auf einzelnen Segmenten oder Zweigen eines Fieldbus-Busses zu kommunizieren und diese miteinander zu verbinden, um größere Prozeßsteuerungsnetze zu kreieren. Falls gewünscht, können Brückeneinrichtungen eine Umwandlung zwischen verschiedenen Datengeschwindigkeiten und/oder verschiedenen Datensignalgabeformaten, die auf den verschiedenen Segmenten des Busses **34** benutzt werden, bewirken, können Signale verstärken, die zwischen den Segmenten des Busses **34** fließen, können die zwischen den verschiedenen Segmenten des Busses **34** fließenden Signale filtern und nur diejenigen Signale weiterleiten, die dazu bestimmt sind, von einer Einrichtung an einem der Bussegmente empfangen zu werden, mit dem die Brücke gekoppelt ist, und/oder können andere Aktionen ausführen, die notwendig sind, um verschiedene Segmente des Busses **34** miteinander zu koppeln. Brückeneinrichtungen, die Bussegmente miteinander verbinden, die mit unterschiedlichen Geschwindigkeiten arbeiten, müssen Linkmaster-Fähigkeiten auf der langsameren Segmentseite der Brücke haben. Die Hauptrechner **12** und **15**, der PLC **13** und die Steuereinheiten **14** können jeder Typ von Fieldbus-Einrichtung sein, sind aber charakteristisch Linkmaster-Einrichtungen.

**[0037]** Jede der Einrichtungen **12** bis **32** ist imstande, über den Bus **34** zu kommunizieren und, was

wichtig ist, selbständig eine oder mehrere Prozeßsteuerfunktionen auszuführen unter Nutzung von Daten, die durch die Einrichtung von dem Prozeß oder von einer anderen Einrichtung über Kommunikationssignale auf dem Bus **34** gewonnen werden. Fieldbus-Einrichtungen können daher Teile einer Gesamtsteuerungsstrategie direkt implementieren, die in der Vergangenheit von einer zentralen digitalen Steuereinheit eines DCS ausgeführt wurden. Zur Durchführung von Steuerfunktionen weist jede Fieldbus-Einrichtung einen oder mehrere standardisierte "Blöcke" auf, die in einem Mikroprozessor innerhalb der Einrichtung implementiert sind. Insbesondere weist jede Fieldbus-Einrichtung einen Ressourcenblock, null oder mehr Funktionsblöcke und null oder mehr Wandlerblöcke auf. Diese Blöcke sind als Blockobjekte bezeichnet.

**[0038]** Ein Ressourcenblock speichert und überträgt einrichtungsspezifische Daten, die sich auf einige der Charakteristiken einer Fieldbus-Einrichtung beziehen, beispielsweise einen Einrichtungstyp, eine Einrichtungs-Revisionsangabe und Angaben darüber, wo andere einrichtungsspezifische Information innerhalb eines Speichers der Einrichtung erhalten werden kann. Zwar können Einrichtungen verschiedener Hersteller unterschiedliche Arten von Daten in dem Ressourcenblock eines Feldgeräts speichern, aber jedes Feldgerät, das mit dem Fieldbus-Protokoll konform ist, enthält einen Ressourcenblock, in dem einige Daten gespeichert sind.

**[0039]** Ein Funktionsblock definiert und implementiert eine Eingangsfunktion, eine Ausgangsfunktion oder eine Steuerfunktion, die der Feldeinrichtung zugeordnet ist, und deshalb werden Funktionsblöcke im allgemeinen als Eingangs-, Ausgangs- und Steuerfunktionsblöcke bezeichnet. Es können jedoch auch andere Kategorien von Funktionsblöcken wie etwa Hybridfunktionsblöcke existieren oder zukünftig entwickelt werden. Jeder Eingangs- oder Ausgangsfunktionsblock erzeugt mindestens einen Prozeßsteuereingang (wie etwa eine Prozeßvariable von einer Prozeßmeßeinrichtung) oder Prozeßsteuerausgang (wie etwa eine Ventilposition, die einer Betätigungseinheit zugeführt wird), während jeder Steuerfunktionsblock einen Algorithmus (der urheberrechtlich geschützt sein kann) verwendet, um einen oder mehrere Prozeßausgänge aus einem oder mehreren Prozeßeingängen und Steuereingängen zu erzeugen. Beispiele von Standard-Funktionsblöcken umfassen Analogeingangs- bzw. AI-, Analogausgangs- bzw. AO-, Bias- bzw. B-, Steuerselektor- bzw. CS-, diskrete Eingangs- bzw. DI-, diskrete Ausgangs- bzw. DO-, manuelle Lade- bzw. ML-, Proportional/Differential- bzw. PD-, Proportional/Integral/Differential- bzw. PID-, Verhältnis- bzw. RA- und Signalselektor- bzw. SS-Funktionsblöcke. Es existieren jedoch andere Arten von Funktionsblöcken, und neue Funktionsblockarten können definiert oder geschaffen werden, um in

der Fieldbus-Umgebung wirksam zu sein.

**[0040]** Ein Wandlerblock koppelt die Eingänge und Ausgänge eines Funktionsblocks mit lokalen Hardwareeinrichtungen wie etwa Sensoren und Gerätebetätigern, um es Funktionsblöcken zu ermöglichen, die Ausgänge von lokalen Sensoren zu lesen und lokale Einrichtungen anzuweisen, eine oder mehrere Funktionen wie etwa das Bewegen eines Ventilelements auszuführen. Wandlerblöcke enthalten typischerweise Information, die erforderlich ist, um Signale zu interpretieren, die von einem lokalen Gerät geliefert werden, und um lokale Hardwareeinrichtungen richtig zu steuern, was beispielsweise Information, die den Typ eines lokalen Gerätes identifiziert, Kalibrierinformation, die einem lokalen Gerät zugeordnet ist, usw. umfaßt. Typischerweise ist jedem Eingangs- oder Ausgangsfunktionsblock ein einzelner Wandlerblock zugeordnet.

**[0041]** Die meisten Funktionsblöcke sind imstande, Alarm- oder Ereignisanzeigen auf der Basis von vorbestimmten Kriterien zu erzeugen und können in verschiedenen Moden unterschiedlich wirksam sein. Allgemein gesagt, können Funktionsblöcke wirksam sein in einem Automatikmodus, in dem beispielsweise der Algorithmus eines Funktionsblocks automatisch wirksam ist; in einem Bedienermodus, in dem der Eingang oder Ausgang eines Funktionsblocks manuell gesteuert wird; in einem Außer-Betrieb-Modus, in dem der Block unwirksam ist; in einem Kaskadenmodus, in dem die Operation des Blocks durch den Ausgang eines anderen Blocks beeinflusst (bestimmt) wird; und in einem oder mehreren Fernmoden, in denen ein entfernter Computer den Modus des Blocks bestimmt. In dem Fieldbus-Protokoll gibt es jedoch weitere Operationsmoden.

**[0042]** Es ist wichtig, daß jeder Block fähig ist, mit anderen Blöcken in denselben oder anderen Feldeinrichtungen über den Fieldbus-Bus **34** zu kommunizieren, und zwar unter Anwendung von Standard-Nachrichtenformaten, die durch das Fieldbus-Protokoll definiert sind. Somit können Kombinationen von Funktionsblöcken (in derselben Einrichtung oder in verschiedenen Einrichtungen) miteinander kommunizieren, um eine oder mehrere dezentrale Steuerschleifen zu bilden. Beispielsweise kann ein PID-Funktionsblock in einer Feldeinrichtung über den Bus **34** so eingekoppelt sein, daß er einen Ausgang eines AI-Funktionsblocks in einer zweiten Feldeinrichtung empfängt, Daten an einen AO-Funktionsblock in einer dritten Feldeinrichtung abgibt und einen Ausgang des AO-Funktionsblocks als Rückführung empfängt, um separat und gesondert von irgendeiner DCS-Steuereinheit eine Prozeßsteuerschleife zu kreieren. Auf diese Weise bewegen Kombinationen von Funktionsblöcken Steuerfunktionen aus einer zentralisierten DCS-Umgebung hinaus, was es DCS-Multifunktionssteuereinheiten erlaubt, überwachende

oder koordinierende Funktionen auszuüben oder vollständig zu entfallen. Außerdem liefern Funktionsblöcke eine grafische, blockorientierte Struktur zur einfachen Konfigurierung eines Prozesses und ermöglichen die Verteilung von Funktionen unter Feldeinrichtungen, die von verschiedenen Herstellern stammen, weil diese Blöcke ein gleichbleibendes Kommunikationsprotokoll nutzen.

**[0043]** Jede Feldeinrichtung enthält und implementiert nicht nur Blockobjekte, sondern weist zusätzlich eines oder mehrere andere Objekte einschließlich Koppelobjekte, Trendobjekte, Warnobjekte und Betrachtungsobjekte auf. Koppelobjekte definieren die Verbindungen bzw. Kopplungen zwischen den Eingängen und Ausgängen von Blöcken (wie etwa Funktionsblöcken) sowohl innerhalb der Feldeinrichtung als auch über den Feldbus-Bus **34**.

**[0044]** Trendobjekte gestatten die lokale Trendbildung von Funktionsblockparametern für den Zugriff durch andere Einrichtungen wie etwa den Hauptrechner **12** oder Steuereinheiten **14** von **Fig. 1**. Trendobjekte enthalten historische Kurzzeitdaten, die irgendeinen Parameter, z. B. einen Funktionsblockparameter betreffen, und berichten diese Daten über den Bus **34** auf asynchrone Weise an andere Einrichtungen oder Funktionsblöcke. Warnobjekte berichten Alarme und Ereignisse über den Bus **34**. Diese Alarme oder Ereignisse können sich auf jedes Ereignis beziehen, das innerhalb einer Einrichtung oder in einem der Blöcke einer Einrichtung stattfindet. Betrachtungsobjekte sind vordefinierte Gruppierungen von Blockparametern, die bei Mensch-Maschine-Standardschnittstellen verwendet werden, und können von Zeit zu Zeit an andere Einrichtungen zum Betrachten übermittelt werden.

**[0045]** **Fig. 2** zeigt drei Feldbus-Einrichtungen, die beispielsweise irgendwelche der Feldgeräte **16** bis **28** von **Fig. 1** sein können, die Ressourcenblöcke **48**, Funktionsblöcke **50**, **51** oder **52** und Wandlerblöcke **53** und **54** aufweisen. Bei der ersten Einrichtung ist der Funktionsblock **50** (der ein Eingangsfunktionsblock sein kann) durch den Wandlerblock **53** mit einem Sensor **55** gekoppelt, der beispielsweise ein Temperatursensor, ein Sollwertanzeigesensor usw. sein kann. In der zweiten Einrichtung ist der Funktionsblock **51** (der ein Ausgangsfunktionsblock sein kann) durch den Wandlerblock **54** mit einem Ausgangsgerät wie etwa einem Ventil **56** verbunden. In der dritten Einrichtung hat der Funktionsblock **52** (der ein Steuerfunktionsblock sein kann) ein zugeordnetes Trendobjekt **57** für die Trendbildung des Eingangsparameters des Funktionsblocks **52**.

**[0046]** Verknüpfungsobjekte **58** definieren die Blockparameter jedes der zugehörigen Blöcke, und Warnobjekte **59** liefern Alarme oder Ereignismeldungen für jeden der zugehörigen Blöcke. Betrachtungs-

objekte **60** sind jedem der Funktionsblöcke **50**, **51** und **52** zugeordnet und umfassen oder gruppieren Datenlisten für die Funktionsblöcke, mit denen sie zusammenwirken. Diese Listen enthalten Information, die für jede einer Gruppe von verschiedenen definierten Ansichten notwendig ist. Selbstverständlich sind die Einrichtungen von **Fig. 2** nur beispielhaft, und andere Anzahlen und Arten von Blockobjekten, Verknüpfungsobjekten, Warnobjekten, Trendobjekten und Betrachtungsobjekten können in jedem Feldgerät vorgesehen sein.

**[0047]** Unter Bezugnahme auf **Fig. 3** zeigt ein Blockbild des Prozeßsteuerungsnetzes **10**, in dem die Geräte **16**, **18** und **24** als Positionierer-/Ventilgeräte und die Geräte **20**, **22**, **26** und **28** als Meßwandler dargestellt sind, außerdem die Funktionsblöcke, die dem Positionierer/Ventil **16**, dem Wandler **20** und der Brücke **30** zugeordnet sind. Wie **Fig. 3** zeigt, umfaßt das Positionierer/Ventil **16** einen Ressourcenblock bzw. RSC-Block **61**, einen Wandler- bzw. XD-CR-Block **62** und eine Reihe von Funktionsblöcken, und zwar einen Analogausgangs- bzw. AO-Funktionsblock **63**, zwei PID-Funktionsblöcke **64** und **65** sowie einen Signalwähl- bzw. SS-Funktionsblock **69**. Der Wandler **20** umfaßt einen Ressourcenblock **61**, zwei Wandlerblöcke **62** und zwei Analogeingangs- bzw. AI-Funktionsblöcke **66** und **67**. Ferner weist die Brücke **30** einen Ressourcenblock **61** und einen PID-Funktionsblock **68** auf.

**[0048]** Es versteht sich, daß die verschiedenen Funktionsblöcke von **Fig. 3** in einer Anzahl von Steuerschleifen zusammenwirken können (indem sie über den Bus **34** kommunizieren), und die Steuerschleifen, in denen Funktionsblöcke des Positionierers/Ventils **16**, des Wandlers **20** und der Brücke **30** liegen, sind in **Fig. 3** durch einen Schleifenerkennungsblock bezeichnet, der mit jedem dieser Funktionsblöcke verbunden ist. Wie also **Fig. 3** zeigt, sind der AO-Funktionsblock **63** und der PID-Funktionsblock **64** des Positionierers/Ventils **16** und der AI-Funktionsblock **66** des Wandlers **20** innerhalb einer Steuerschleife LOOP1 miteinander verbunden, wogegen der SS-Funktionsblock **69** des Positionierers/Ventils **16**, der AI-Funktionsblock **67** des Wandlers **20** und der PID-Funktionsblock **68** der Brücke **30** in einer Steuerschleife LOOP2 miteinander verbunden sind. Der andere PID-Funktionsblock **65** des Positionierers/Ventils **16** ist in eine Steuerschleife LOOP3 eingekoppelt.

**[0049]** Die miteinander verbundenen Funktionsblöcke, welche die Steuerschleife LOOP1 in **Fig. 3** bilden, sind im einzelnen in dem Schema dieser Steuerschleife gemäß **Fig. 4** dargestellt. Wie **Fig. 4** zeigt, ist die Steuerschleife LOOP1 vollständig durch Kommunikationsverknüpfungen zwischen dem AO-Funktionsblock **63** und dem PID-Funktionsblock **64** des Positionierers/Ventils **16** und dem AI-Funktionsblock

**66** des Wandlers **20** ([Fig. 3](#)) gebildet. Das Steuer-schleifenschema von [Fig. 4](#) zeigt die Kommunikationsverbindungen zwischen diesen Funktionsblöcken unter Verwendung von Leitungen zur Verbindung der Prozeß- und Steuereingänge und -ausgänge dieser Funktionsblöcke. So ist der Ausgang des AI-Funktionsblocks **66**, der ein Prozeßmeß- oder Prozeßparametersignal aufweisen kann, kommunikativ über das Bussegment **34b** mit dem Eingang des PID-Funktionsblocks **64** gekoppelt, der einen Ausgang hat, der ein Steuersignal aufweist, das kommunikativ mit einem Eingang des AO-Funktionsblocks **63** gekoppelt ist. Ein Ausgang des AO-Funktionsblocks **63**, der ein Rückführungssignal aufweist, das beispielsweise die Position des Ventils **16** bezeichnet, ist mit einem Steuereingang des PID-Funktionsblocks **64** verbunden. Der PID-Funktionsblock **64** nutzt dieses Rückführungssignal gemeinsam mit dem Prozeßmeßsignal von dem AI-Funktionsblock **66** zur Implementierung einer ordnungsgemäßen Steuerung des AO-Funktionsblocks **63**. Selbstverständlich können die durch die Leitungen in dem Steuer-schleifenschema von [Fig. 4](#) gezeigten Verbindungen innerhalb eines Feldeingeräts intern ausgeführt werden, wenn wie im Fall der AO- und der PID-Funktionsblöcke **63** und **64** die Funktionsblöcke innerhalb desselben Feldgeräts liegen (z. B. des Positionierers/Ventils **16**), oder diese Verbindungen können über den Zweidraht-Kommunikationsbus **34** unter Anwendung von synchronen Standard-Feldbus-Kommunikationen implementiert werden. Natürlich werden von anderen Funktionsblöcken, die in anderen Konfigurationen kommunikativ gekoppelt sind, andere Steuer-schleifen implementiert.

**[0050]** Zur Implementierung und Durchführung von Kommunikations- und Steueraktivitäten verwendet das Feldbus-Protokoll drei allgemeine technologische Kategorien, die als eine physikalische Schicht, ein Kommunikations-"Stapel" und eine Anwenderschicht bezeichnet sind. Die Anwenderschicht umfaßt die Steuerungs- und Konfigurationsfunktionen, die in Form von Blöcken (wie etwa Funktionsblöcken) und Objekten innerhalb jeder einzelnen Prozeßsteuer-einrichtung oder jedes Feldgeräts vorgesehen sind. Die Anwenderschicht ist typischerweise vom Hersteller der Einrichtung auf eine gesetzlich geschützte Weise ausgebildet, muß jedoch imstande sein, Nachrichten entsprechend dem durch das Feldbus-Protokoll definierten Standardnachrichtenformat zu empfangen und zu senden, und muß von einem Anwender auf eine dem Standard entsprechende Weise konfiguriert werden können. Die physikalische Schicht und der Kommunikationsstapel sind notwendig, um die Kommunikation zwischen verschiedenen Blöcken verschiedener Feldgeräte auf standardisierte Weise unter Verwendung des Zweidraht-Busses **34** zu bewirken, und kann nach dem wohlbekanntem OSI-Schichtkommunikationsmodell ausgebildet sein.

**[0051]** Die physikalische Schicht, die der OSI-Schicht **1** entspricht, ist in jedes Feldgerät und den Bus **34** eingebettet und ist wirksam zur Umwandlung von elektromagnetischen Signalen, die von dem Feldbus-Übertragungsmedium (dem Zweidraht-Bus **34**) empfangen werden, in Nachrichten, die von dem Kommunikationsstapel der Feldeinrichtung genutzt werden können. Die physikalische Schicht kann man sich als den Bus **34** und die elektromagnetischen Signale vorstellen, die auf dem Bus **34** an den Eingängen und Ausgängen der Feldgeräte vorhanden sind.

**[0052]** Der Kommunikationsstapel, der in jeder Feldbus-Einrichtung vorhanden ist, umfaßt eine Datensicherungsschicht, die der OSI-Schicht **2** entspricht, eine Feldbus-Zugriffsunterschicht und eine Feldbus-Nachrichtendarstellungsschicht, die der OSI-Schicht **6** entspricht. In dem Feldbus-Protokoll gibt es keine entsprechende Struktur für die OSI-Schichten **3** bis **5**. Die Anwendungen einer Feldbus-Einrichtung weisen jedoch eine Schicht **7** auf, wogegen eine Anwenderschicht eine Schicht **8** ist, die in dem OSI-Protokoll nicht definiert ist. Jede Schicht in dem Kommunikationsstapel ist zuständig für das Codieren oder Decodieren eines Anteils der Nachricht oder des Signals, das auf dem Feldbus-Bus **34** übertragen wird. Infolgedessen addiert oder entfernt jede Schicht des Kommunikationsstapels bestimmte Anteile des Feldbus-Signals wie etwa Präambeln, Startbegrenzer und Endbegrenzer und decodiert in manchen Fällen die entfernten Anteile des Feldbus-Signals, um festzustellen, wohin der Rest des Signals oder der Nachricht zu senden ist oder ob das Signal verworfen werden soll, weil es beispielsweise eine Nachricht oder Daten für Funktionsblöcke enthält, die sich nicht in dem empfangenden Feldgerät befinden.

**[0053]** Die Datensicherungsschicht steuert die Übertragung von Nachrichten auf den Bus **34** und verwaltet den Zugriff auf den Bus **34** nach Maßgabe eines deterministischen zentralen Busabwicklers, der als link-aktiver Verteiler bezeichnet und noch im einzelnen erläutert wird. Die Datensicherungsschicht entfernt eine Präambel von den Signalen auf dem Übertragungsmedium und kann die empfangene Präambel nutzen, um den internen Takt des Feldgeräts mit dem ankommenden Feldbus-Signal zu synchronisieren. Ebenso wandelt die Datensicherungsschicht Nachrichten an dem Kommunikationsstapel in physikalische Feldbus-Signale um und codiert diese Signale mit Taktinformation, um ein "synchrones serielles" Signal zu erzeugen, das eine richtige Präambel für die Übertragung auf dem Zweidraht-Bus **34** hat. Während des Decodiervorgangs erkennt die Datensicherungsschicht Spezialcodes innerhalb der Präambel wie etwa Startbegrenzer und Endbegrenzer, um den Beginn und das Ende einer bestimmten Feldbus-Nachricht zu erkennen, und kann eine Prüfsumme ausführen, um die Integrität des Signals oder

der Meldung, die von dem Bus **34** empfangen wird, zu verifizieren. Ebenso überträgt die Datensicherungsschicht Fieldbus-Signale auf dem Bus **34** durch Hinzufügen von Start- und Endbegrenzern zu Nachrichten an dem Kommunikationsstapel und Plazieren dieser Signale zum richtigen Zeitpunkt auf dem Übertragungsmedium.

**[0054]** Die Fieldbus-Nachrichtendarstellungsschicht erlaubt der Anwenderschicht (d. h. den Funktionsblöcken, Objekten usw. eines Feldgeräts), unter Anwendung einer Standardmenge von Nachrichtenformaten über den Bus **34** zu kommunizieren, und beschreibt die Kommunikationsdienste, Nachrichtenformate und das Protokollverhalten, die notwendig sind, um Nachrichten aufzubauen, die auf dem Kommunikationsstapel zu plazieren und der Anwenderschicht verfügbar zu machen sind. Da die Fieldbus-Nachrichtendarstellungsschicht standardisierte Kommunikationen für die Anwenderschicht liefert, sind für jeden Typ von Objekt, der oben beschrieben wurde, spezielle Fieldbus-Nachrichtendarstellungs-Kommunikationsdienste definiert. Beispielsweise weist die Fieldbus-Nachrichtendarstellungsschicht Objektwörterbuchdienste auf, was es einem Anwender ermöglicht, ein Objektwörterbuch eines Geräts zu lesen. Das Objektwörterbuch speichert Objektbeschreibungen, die jedes der Objekte (wie etwa Blockobjekte) eines Geräts beschreiben oder identifizieren. Die Fieldbus-Nachrichtendarstellungsschicht stellt außerdem Kontextverwaltungsdienste bereit, was es einem Anwender erlaubt, Kommunikationsbeziehungen zu lesen und zu ändern, die als noch zu beschreibende virtuelle Kommunikationsbeziehungen bzw. VCR bekannt sind und einem oder mehreren Objekten einer Einrichtung zugeordnet sind. Ferner bietet die Fieldbus-Nachrichtendarstellungsschicht variable Zugriffsdienste, Ereignisdienste, Hoch- und Herunterladedienste und Programmaufrufdienste, die sämtlich in dem Fieldbus-Protokoll wohlbekannt sind und daher hier nicht im einzelnen beschrieben werden. Die Fieldbus-Zugriffsschicht bildet die Fieldbus-Nachrichtendarstellungsschicht in die Datensicherungsschicht ab.

**[0055]** Um die Operation dieser Schichten zuzulassen oder zu aktivieren, enthält jede Fieldbus-Einrichtung eine Managementinformationsbasis bzw. MIB, die eine Datenbank ist, in der VCR, dynamische Variablen, Statistiken, Zeitabläufe von link-aktiven Verteilern, Funktionsblock-Ausführungszeitpläne und Gerätekennungs- und -adreßinformationen gespeichert sind. Selbstverständlich kann auf die Informationen innerhalb der MIB jederzeit unter Anwendung von Standard-Fieldbus-Nachrichten oder -Befehlen Zugriff erfolgen oder können Änderungen vorgenommen werden. Ferner wird gewöhnlich mit jedem Gerät eine Gerätebeschreibung zur Verfügung gestellt, um einem Anwender oder einem Hauptrechner ein erweitertes Bild der Information in dem VFD zu ge-

ben. Eine Gerätebeschreibung, die typischerweise mit einer Markierung bzw. einem Token versehen sein muß, um von einem Hauptrechner verwendet werden zu können, speichert Informationen, die erforderlich sind, damit der Hauptrechner die Bedeutung der Daten in den VFD einer Einrichtung verstehen kann.

**[0056]** Es versteht sich, daß zur Implementierung einer Steuerungsstrategie unter Verwendung von Funktionsblöcken, die in einem gesamten Prozeßsteuerungsnetz verteilt sind, die Ausführung der Funktionsblöcke in Bezug auf die Ausführung anderer Funktionsblöcke in einer bestimmten Steuerungsschleife exakt geplant sein muß. Ebenso muß die Kommunikation zwischen verschiedenen Funktionsblöcken auf dem Bus **34** genau geplant sein, damit jeder Funktionsblock die richtigen Daten erhält, bevor dieser Block ausführt.

**[0057]** Die Art und Weise, wie verschiedene Feldgeräte (und verschiedene Blöcke in Feldgeräten) über das Fieldbus-Übertragungsmedium kommunizieren, wird nun unter Bezugnahme auf [Fig. 1](#) beschrieben. Damit eine Kommunikation stattfindet, wirkt eine der Linkmastereinrichtungen an jedem Segment des Busses **34** (z. B. Einrichtungen **12**, **16** und **26**) als link-aktiver Verteiler bzw. LAS, der die Kommunikation auf dem zugehörigen Segment des Busses **34** aktiv plant und steuert. Der LAS für jedes Segment des Busses **34** speichert und aktualisiert einen Kommunikationsplan (eine link-aktive Verteilung), der die Zeiten enthält, zu denen jeder Funktionsblock jedes Geräts planmäßig mit einer periodischen Kommunikationsaktivität auf dem Bus **34** beginnt, und die Zeitdauer, während der diese Kommunikationsaktivität stattfinden soll. Es kann zwar an jedem Segment des Busses **34** eine, und zwar nur eine, aktive LAS-Einrichtung geben, aber andere Linkmastereinrichtungen (wie die Einrichtung **22** an dem Segment **34b**) kann als Sicherungs-LAS dienen und aktiv werden, wenn beispielsweise der aktuelle LAS ausfällt. Basis-einrichtungen haben nicht die Fähigkeit, zu irgendeiner Zeit ein LAS zu werden.

**[0058]** Allgemein gesagt, sind Kommunikationsaktivitäten über den Bus **34** in sich wiederholende Makrozyklen unterteilt, von denen jeder eine synchrone Kommunikation für jeden Funktionsblock, der an einem bestimmten Segment des Busses **34** aktiv ist, und eine oder mehrere asynchrone Kommunikationen für einen oder mehrere der Funktionsblöcke oder Geräte, die an einem Segment des Busses **34** aktiv sind, aufweist. Eine Einrichtung kann aktiv sein, d. h. Daten zu einem Segment des Busses **34** senden und solche davon empfangen, auch wenn sie physisch mit einem anderen Segment des Busses **34** verbunden ist, und zwar durch eine koordinierte Operation der Brücken und der LAS an dem Bus **34**.

**[0059]** Während jedes Makrozyklus führt jeder an einem bestimmten Segment des Busses **34** aktive Funktionsblock gewöhnlich zu einer anderen, jedoch präzise geplanten (synchronen) Zeit aus, und zu einer anderen präzise geplanten Zeit publiziert er seine Ausgangsdaten an diesem Segment des Busses **34** als Reaktion auf einen Zwangs-Datenbefehl, der von dem entsprechenden LAS erzeugt wird. Bevorzugt publiziert jeder Funktionsblock seine Ausgangsdaten planmäßig kurz nach dem Ende der Ausführungsperiode des Funktionsblocks. Außerdem sind die Datenpublizierungszeiten der verschiedenen Funktionsblöcke seriell geplant, so daß keine zwei Funktionsblöcke an einem bestimmten Segment des Busses **34** Daten gleichzeitig publizieren. Während der Zeitdauer, in der keine synchrone Kommunikation stattfindet, darf jedes Feldgerät aufeinanderfolgend Alarmdaten, Betrachtungsdaten usw. auf asynchrone Weise unter Nutzung von tokengesteuerten Kommunikationen übertragen. Die Ausführungszeiten und die erforderliche Zeitdauer zur Fertigstellung der Ausführung jedes Funktionsblocks sind in der Managementinformationsbasis bzw. MIB der Einrichtung gespeichert, in der sich der Funktionsblock befindet, wogegen, wie erwähnt, die Zeitpunkte zum Senden der Zwangs-Datenbefehle an jede der Einrichtungen an einem Segment des Busses **34** in der MIB der LAS-Einrichtung für dieses Segment gespeichert sind. Diese Zeiten sind typischerweise als Offsetzeiten gespeichert, weil sie die Zeiten, zu denen ein Funktionsblock ausführen oder Daten senden soll, als einen Versatz bzw. Offset vom Beginn einer "absoluten Linkplanstartzeit" identifizieren, die allen mit dem Bus **34** verbundenen Einrichtungen bekannt ist.

**[0060]** Zur Durchführung von Kommunikationen während jedes Makrozyklus sendet der LAS, beispielsweise der LAS **16** des Bussegments **34b**, einen Zwangs-Datenbefehl an jede der Einrichtungen an dem Bussegment **34b** entsprechend der Liste von Sendezeiten, die in dem link-aktiven Plan gespeichert sind. Bei Empfang eines Zwangs-Datenbefehls publiziert ein Funktionsblock einer Einrichtung seine Ausgangsdaten für eine bestimmte Zeitdauer auf dem Bus **34**. Da die Ausführung jedes der Funktionsblöcke typischerweise so geplant ist, daß die Ausführung dieses Blocks beendet ist, kurz bevor der Block planmäßig einen Zwangs-Datenbefehl empfängt, sollten die als Reaktion auf einen Zwangs-Datenbefehl publizierten Daten die neuesten Ausgangsdaten des Funktionsblocks sein. Wenn jedoch ein Funktionsblock langsam ausführt und keine neuen Ausgänge zwischengespeichert hat, wenn er den Zwangs-Datenbefehl empfängt, publiziert der Funktionsblock die während des letzten Durchlaufs des Funktionsblocks erzeugten Ausgangsdaten und zeigt unter Verwendung eines Zeitstempels an, daß die publizierten Daten alte Daten sind.

**[0061]** Nachdem der LAS an jeden der Funktionsblöcke an einem bestimmten Segment des Busses **34** einen Zwangs-Datenbefehl gesendet hat und während der Zeiten, in denen Funktionsblöcke ausführen, kann der LAS veranlassen, daß asynchrone Kommunikationsaktivitäten stattfinden. Zur Ausführung einer asynchronen Kommunikation sendet der LAS eine Tokenpassiernachricht an ein bestimmtes Feldgerät. Wenn ein Feldgerät eine Tokenpassiernachricht empfängt, hat dieses Feldgerät vollständigen Zugang zu dem Bus **34** (oder einem Segment davon) und kann asynchrone Nachrichten wie Alarmmeldungen, Trenddaten, vom Bediener eingestellte Sollwertänderungen usw. senden, bis die Meldungen komplett sind oder bis eine maximal zugewiesene "Tokenhaltezeit" abgelaufen ist. Danach gibt das Feldgerät den Bus (oder ein bestimmtes Segment davon) frei, und der LAS sendet eine Tokenpassiernachricht an ein anderes Gerät. Dieser Prozeß wird bis zum Ende des Makrozyklus wiederholt oder bis der LAS planmäßig einen Zwangs-Datenbefehl senden muß, um eine synchrone Kommunikation zu bewirken. Selbstverständlich kann während jedes Makrozyklus nicht jedes Gerät eine Tokenpassiernachricht empfangen, was von dem Nachrichtenverkehrsaufkommen und der Anzahl Geräte und Blöcke abhängig ist, die mit einem bestimmten Segment des Busses **34** verbunden sind.

**[0062]** [Fig. 5](#) zeigt einen Zeitplan, der die Zeiten zeigt, zu denen Funktionsblöcke an dem Bussegment **34b** von [Fig. 1](#) während jedes Makrozyklus des Bussegments **34b** ausführen, und die Zeiten zeigt, zu denen synchrone Kommunikationen während jedes dem Bussegment **34b** zugeordneten Makrozyklus stattfinden. In dem Zeitablaufplan von [Fig. 5](#) ist die Zeit auf der Horizontalachse aufgetragen, und zu den verschiedenen Funktionsblöcken des Positionierers/Ventils **16** und des Wandlers **20** (von [Fig. 3](#)) gehörende Aktivitäten sind auf der Vertikalachse aufgetragen. Die Steuerschleife, in der jeder der Funktionsblöcke wirksam ist, ist in [Fig. 5](#) als Indexbezeichnung angegeben. So bezieht sich  $AI_{LOOP1}$  auf den AI-Funktionsblock **66** des Wandlers **20**,  $PID_{LOOP1}$  bezieht sich auf den PID-Funktionsblock **64** des Positionierers/Ventils **16** usw. Die Blockausführungszeit jedes gezeigten Funktionsblocks ist durch ein schraffiertes Kästchen bezeichnet, während jede planmäßige synchrone Kommunikation durch einen vertikalen Balken in [Fig. 5](#) bezeichnet ist.

**[0063]** Nach dem Zeitablaufplan von [Fig. 5](#) führt also während eines bestimmten Makrozyklus des Segments **34b** ([Fig. 1](#)) der Funktionsblock  $AI_{LOOP1}$  zuerst für den durch das Kästchen **70** bezeichneten Zeitraum aus. Dann wird während des Zeitraums, der durch den vertikalen Balken **72** bezeichnet ist, der Ausgang des Funktionsblocks  $AI_{LOOP1}$  auf dem Bussegment **34b** publiziert als Reaktion auf einen Zwangs-Datenbefehl von dem LAS für das Busseg-

ment **34b**. Ebenso bezeichnen die Kästchen **74**, **76**, **78**, **80** und **81** die Ausführungszeiten der Funktionsblöcke  $PID_{LOOP1}$ ,  $AI_{LOOP2}$ ,  $AO_{LOOP1}$ ,  $SS_{LOOP2}$  bzw.  $PID_{LOOP3}$  (die für jeden der verschiedenen Blöcke verschieden sind), während die vertikalen Balken **82**, **84**, **86**, **88** und **89** die Zeiten angeben, zu denen die Funktionsblöcke  $PID_{LOOP1}$ ,  $AI_{LOOP2}$ ,  $AO_{LOOP1}$ ,  $SS_{LOOP2}$  bzw.  $PID_{LOOP3}$  Daten auf dem Bussegment **34b** publizieren.

**[0064]** Es ist ersichtlich, daß der Zeitablaufplan von **Fig. 5** auch die Zeitdauern zeigt, die für asynchrone Kommunikationsaktivitäten verfügbar sind und stattfinden können während der Ausführungszeiten jedes der Funktionsblöcke und während der Zeitdauer am Ende des Makrozyklus, in der keine Funktionsblöcke ausführen, und wenn auf dem Bussegment **34b** keine synchrone Kommunikation stattfindet. Falls gewünscht, können natürlich andere Funktionsblöcke bewußt so eingeplant sein, daß sie gleichzeitig ausführen, und nicht alle Funktionsblöcke müssen Daten auf dem Bus publizieren, wenn beispielsweise keine andere Einrichtung an den von einem Funktionsblock erzeugten Daten teilnimmt.

**[0065]** Feldgeräte können Daten und Nachrichten über den Bus **34** publizieren oder übertragen, indem sie eine von drei virtuellen Kommunikationsbeziehungen (VCR) nutzen, die in der Fieldbus-Zugriffsunterschicht des Stapels jedes Feldgeräts definiert sind. Eine Client/Server-VCR wird für in Warteschlange befindliche, ungeplante, vom Anwender ausgelöste Eins-zu-Eins-Kommunikationen zwischen Geräten an dem Bus **34** verwendet. Diese Warteschlangennachrichten werden in der Reihenfolge, in der sie zur Übertragung geliefert wurden, nach ihrer Priorität und ohne Überschreiben vorhergehender Nachrichten gesendet und empfangen. So kann ein Feldgerät eine Client/Server-VCR nutzen, wenn es eine Tokenpassiermeldung von einem LAS empfängt, um eine Anforderungsnachricht an ein anderes Gerät an dem Bus **34** zu senden. Der Anforderer wird als "Client" bezeichnet, und das die Anforderung empfangende Gerät wird als "Server" bezeichnet. Der Server sendet eine Antwort, wenn er eine Tokenpassiermeldung von dem LAS empfängt. Die Client/Server-VCR wird beispielsweise angewandt, um vom Bediener ausgelöste Anforderungen wie etwa Sollwertänderungen, Zugang zu Abstimmparametern und Änderungen, Alarmbestätigungen und Geräte-Uploads und -Downloads auszuführen.

**[0066]** Eine Berichtverteilungs-VCR wird für in Warteschlange befindliche, ungeplante, vom Anwender ausgelöste Eins-an-Viele-Kommunikationen verwendet. Wenn beispielsweise ein Feldgerät mit einem Ereignis- oder einem Trendbericht ein Passiertoken von einem LAS empfängt, sendet dieses Feldgerät seine Nachricht an eine "Gruppenadresse", die in der Fieldbus-Zugriffsunterschicht des Kommunikationsstapels

dieses Geräts definiert ist. Geräte, die so konfiguriert sind, daß sie an dieser VCR teilnehmen, empfangen den Bericht. Der Berichtverteilungs-VCR-Typ wird charakteristisch von Fieldbus-Geräten genutzt, um Alarminformationen an Bedienerkonsolen zu senden.

**[0067]** Ein Publisher/Teilnehmer-VCR-Typ wird für zwischengespeicherte Eins-an-Viele-Kommunikationen verwendet. Zwischengespeicherte Kommunikationen sind solche, die nur die neueste Version der Daten speichern und senden, und daher werden vorhergehende Daten von neuen Daten vollständig überschrieben. Funktionsblockausgänge weisen beispielsweise zwischengespeicherte Daten auf. Ein "Publisher"-Feldgerät publiziert oder sendet eine Nachricht unter Nutzung des Publisher/Teilnehmer-VCR-Typs an alle "Teilnehmer"-Feldgeräte an dem Bus **34**, wenn das Publisher-Gerät eine Zwangs-Datennachricht von dem LAS oder von einem Teilnehmergerät empfängt. Die Publisher/Teilnehmer-Beziehungen sind vorbestimmt und in der Fieldbus-Zugriffsunterschicht des Kommunikationsstapels jedes Feldgeräts definiert und gespeichert.

**[0068]** Um die richtigen Kommunikationsaktivitäten über den Bus **34** zu gewährleisten, sendet jeder LAS periodisch eine Zeitverteilungsmeldung an alle Feldgeräte, die mit einem Segment des Busses **34** verbunden sind, was es den empfangenden Geräten ermöglicht, ihre lokale Anwendungszeit so einzustellen, daß sie miteinander synchron sind. Zwischen diesen Synchronisierungsmeldungen wird in jedem Gerät auf der Basis seines eigenen internen Takts die Taktzeit jeweils unabhängig aufrechterhalten. Die Taktsynchronisierung erlaubt den Feldgeräten, Daten im gesamten Fieldbus-Netzwerk mit Zeitstempel zu versehen, um beispielsweise anzuzeigen, wann Daten erzeugt wurden.

**[0069]** Ferner speichert jeder LAS (und jede andere Linkmastereinrichtung) an jedem Bussegment eine "aktuelle Liste", die eine Liste aller Geräte ist, die mit diesem Segment des Busses **34** verbunden sind, d. h. sämtlicher Geräte, die auf eine Tokenpassiernachricht richtig reagieren. Der LAS erkennt ständig neue Geräte, die einem Bussegment hinzugefügt werden, durch periodisches Senden von Prüfknotenmeldungen an Adressen, die sich nicht auf der aktuellen Liste befinden. Tatsächlich muß jeder LAS mindestens eine Adresse prüfen, nachdem er einen Zyklus des Sendens von Tokenpassiernachrichten an alle Feldgeräte in der aktuellen Liste beendet hat. Wenn an der geprüften Adresse ein Feldgerät vorhanden ist und die Prüfknotennachricht empfängt, sendet das Gerät sofort eine Prüfantwortmeldung zurück. Bei Empfang einer Prüfantwortmeldung fügt der LAS das Gerät der aktuellen Liste hinzu und bestätigt durch Senden einer Knotenaktivierungsmeldung an das geprüfte Feldgerät. Ein Feldgerät bleibt so lange auf der aktuellen Liste, wie dieses Feldgerät auf Tokendurch-

laßmeldungen richtig anspricht. Ein LAS entfernt jedoch ein Feldgerät von der aktuellen Liste, wenn das Feldgerät nach drei aufeinanderfolgenden Versuchen entweder das Token nicht nutzt oder das Token sofort zu dem LAS zurücksendet. Wenn ein Feldgerät zu der aktuellen Liste hinzugefügt oder von ihr entfernt wird, sendet der LAS Änderungen der aktuellen Liste an alle anderen Linkmastereinrichtungen an dem entsprechenden Segment des Busses **34**, damit jede Linkmastereinrichtung eine aktuelle Kopie der aktuellen Liste unterhalten kann.

**[0070]** Wie oben gesagt wurde, werden die Kommunikationsverbindungen zwischen den Feldgeräten und ihren Funktionsblöcken von einem Anwender bestimmt und innerhalb des Prozeßsteuerungsnetzes **10** implementiert unter Anwendung einer Konfigurationsanwendung, die sich beispielsweise in dem Hauptrechner **12** befindet. Nach dem Konfigurieren ist jedoch das Prozeßsteuerungsnetz **10** ohne jede Berücksichtigung von Gerät- oder Prozeßdiagnosevorgängen wirksam und ist daher mit dem Hauptrechner **12** im Dialog, um zwar Standard-E/A-Funktionen, jedoch keine Diagnosefunktionen auszuführen.

**[0071]** Unter Bezugnahme auf [Fig. 6](#) zeigt ein schematisches Blockbild ein Steuersystemnetz **200**, in dem die Redundanz gemäß der vorliegenden Erfindung selektiv implementiert ist unter Verwendung eines einzigen Sets von Kommunikationsmedien in einer einzigen Kommunikationsschleife **202** mit redundanten funktionalen Elementen, die ein primäres Feldgerät **204** und ein redundantes Feldgerät **206** aufweisen. Eine Schleifensteuerung **208** wie etwa eine DCS-Steuereinheit oder ein Feldgerät ist mit der einzigen Kommunikationsschleife **202** verbunden, und die einzige Kommunikationsschleife **202** ist mit dem redundanten Paar von Feldgeräten **204** und **206** gekoppelt. Die Feldgeräte **204** und **206** detektieren und kommunizieren fakultativ einen Ausfallstatus. Die Schleifensteuerung **208** überwacht kontinuierlich die Operation von Geräten in dem Steuersystemnetz **200** unter Anwendung von digitalen Zweiwege-Kommunikationen und detektiert einen Abbruch von Kommunikationen von einem fehlerhaften Feldgerät. Das Steuersystemnetz **200** erholt sich von einem Ausfall des primären Feldgeräts **204**, jedoch nicht vom Ausfall des einzigen Sets von Kommunikationsmedien **202**. Die Schleifensteuerung **208** steuert die Redundanz der Operation der einzelnen Kommunikationsschleife **202** in Kombination mit redundanten Feldgeräten **204** und **206** und detektiert den Ausfall eines funktionalen Elements entweder durch den Empfang eines Ausfallstatus von einem oder mehreren der funktionalen Elemente wie etwa der Steuerlogik in den Geräten oder durch Detektieren einer Diskontinuität von Meldungen von einem oder mehreren der funktionalen Elemente. Beispielsweise umfaßt ein Feldgerät wie etwa ein Prozeßsteuerventil einen

Sensor und ein Rückkopplungssignal, das den Status des Sensors bezeichnet, der wiederum den Operationsstatus des Prozeßsteuerventils bezeichnet. Der Status der Ventiloperation kann eine Bezeichnung eines Ausfallstatus, eines Betriebsstatus oder eines Status aufweisen, der unterschiedliche Funktionalitätsgrade angibt. Das Prozeßsteuerventil und andere ausgewählte funktionale Elemente nutzen bevorzugt die Zweiwege-Kommunikationen der Kommunikationsschleife **202** zum Übertragen einer Statusmeldung an die Schleifensteuereinheit **208**.

**[0072]** Die Schleifensteuereinheit **208** rekonfiguriert automatisch die Kommunikationsschleife des redundanten Paares durch Deaktivieren eines ausgefallenen oder nachlassenden Geräts wie etwa des primären Feldgeräts **204** und durch Aktivieren des entsprechenden alternativen Geräts, das als das redundante Feldgerät **206** dargestellt ist. Funktionale Elemente können Fühler Elemente wie etwa Geber und Steuerelemente wie etwa Ventile oder Motoren sowie andere Feldgeräte und Steuereinrichtungen in einem Prozeß aufweisen. Im Fall von Fühler Elementen werden Übertragungen von einem ausgefallenen Geber ignoriert. Im Fall eines ausgefallenen Steuerelements gibt die Schleifensteuereinheit einen Befehl aus, um das ausgefallene Steuerelement in einen sicheren Betriebsmodus zu deaktivieren.

**[0073]** Unter Bezugnahme auf [Fig. 7](#) zeigt ein schematisches Blockbild ein Steuersystemnetz **300**, bei dem eine Redundanz selektiv implementiert ist unter Verwendung eines redundanten Sets von Kommunikationsmedien einschließlich eines primären Kommunikationsbusses **304** und eines redundanten Kommunikationsbusses **303**, in den redundante Geräte eingekoppelt sind, die ein primäres Feldgerät **304** und ein redundantes Feldgerät **306** aufweisen. Eine Schleifensteuereinheit **308** wie etwa eine DCS-Steuereinheit oder ein Feldgerät ist mit dem primären Kommunikationsbus **302** und dem redundanten Kommunikationsbus **303** verbunden, um ein redundantes Paar von Kommunikationsschleifen zu bilden. Das primäre Feldgerät **304** und das redundante Feldgerät **306** detektieren und kommunizieren fakultativ einen Ausfallstatus. Die Schleifensteuereinheit **308** steuert die Redundanzoperation der redundanten Kommunikationsschleifen durch kontinuierliches Überwachen der Operation von Einrichtungen in dem Steuersystemnetz **300** unter Verwendung von digitalen Zweiwege-Kommunikationen und detektiert den Ausfall eines funktionalen Elements (das ein Bus oder ein Gerät sein kann) entweder durch Empfang eines Ausfallstatus von dem funktionalen Element oder durch Detektieren einer Diskontinuität bzw. Unterbrechung von Meldungen von dem funktionalen Element. Die Schleifensteuereinheit **308** rekonfiguriert automatisch das redundante Paar von Kommunikationsschleifen durch Deaktivieren einer Schleife wie etwa derjenigen, die dem primären Kommunika-

tionsbus **302** zugewiesen oder, oder von einem oder mehreren der Elemente in der primären Schleife bei Ausfall entweder des Busses **302** oder des primären Feldgeräts **304** und durch anschließendes Aktivieren der entsprechenden alternativen Schleife (z. B. derjenigen, die dem redundanten Kommunikationsbus **303** zugewiesen ist) und/oder von einem oder mehreren funktionalen Einrichtungen an dem Bus **303** wie etwa des redundanten Feldgeräts **306**. Dementsprechend wird eine Erholung sowohl für ein ausfallendes funktionales Element als auch für ein ausfallendes Kommunikationsmedium erreicht.

[0074] Unter Bezugnahme auf [Fig. 8](#) zeigt ein schematisches Blockbild ein Steuersystemnetz **400**, bei dem die Redundanz selektiv implementiert ist unter Anwendung eines redundanten Sets von Kommunikationsmedien mit einem primären Kommunikationsbus **402** und einem redundanten Kommunikationsbus **403** mit einem einzigen weiteren funktionalen Element wie etwa einem Feldgerät **404**, das daran gekoppelt ist. Das Feldgerät **404** weist zwei Sets von Schnittstellenelektronik (nicht gezeigt) auf, um das redundante Medium zu nutzen. Eine Schleifensteuereinheit **408** ist mit einem primären Kommunikationsbus **402** und dem redundanten Kommunikationsbus **403** gekoppelt unter Bildung eines redundanten Paares von Kommunikationsschleifen. Das Gerät **404** hat einen ersten Eingangsanschluß und einen ersten Ausgangsanschluß an den primären Kommunikationsbus **402** und hat einen zweiten Eingangsanschluß und einen zweiten Ausgangsanschluß an den redundanten Kommunikationsbus **403**. Somit ist das Gerät **404** in das redundante Paar von Kommunikationsschleifen eingekoppelt. Das einzige Feldgerät **404** detektiert und kommuniziert fakultativ einen Ausfallstatus. Die Schleifensteuereinheit **408** steuert die Redundanzoperation der redundanten Kommunikationsschleifen durch kontinuierliche Überwachung der Operation von Geräten in dem Steuersystemnetz **400** unter Verwendung von digitalen Zweiwege-Kommunikationen und detektiert den Ausfall eines funktionalen Elements entweder durch Empfang eines Ausfallstatus von dem funktionalen Element oder durch Detektieren einer Diskontinuität von Meldungen von dem funktionalen Element. Die Schleifensteuereinheit **408** rekonfiguriert automatisch das redundante Paar von Kommunikationsschleifen durch Deaktivieren eines Busses wie etwa des primären Kommunikationsbusses **402** bei Ausfall des primären Kommunikationsbusses **402** und durch Aktivieren des entsprechenden alternativen redundanten Kommunikationsbusses **403**. Bei dieser Konfiguration ist jedoch die Schleifensteuereinheit **408** nicht fähig, sich von einem Ausfall des Feldgeräts **404** zu erholen. Somit erreicht also das Steuersystemnetz **400**, das ein redundantes Medium, aber ein einzelnes Gerät oder anderes funktionales Element hat, zwar eine Erholung für ein ausfallendes Kommunikationsmedium, gewinnt jedoch die Funktionalität im Fall des aus-

fallenden Geräts oder anderen funktionalen Elements nicht zurück.

[0075] Unter Bezugnahme auf [Fig. 9](#) zeigt ein schematisches Blockbild ein Steuersystemnetz **500** mit zwei funktionalen Elementen, und zwar einem primären Prozeßsteuerventil **502** und einem redundanten Prozeßsteuerventil **504**, die in einen Prozeßdurchflußstrom **512** eingekoppelt sind. In dem Steuersystemnetz **500** sind das primäre Prozeßsteuerventil **502** und das redundante Prozeßsteuerventil **504** in eine einzelne Zweidraht-Kommunikationsschleife **506** eingekoppelt, die von einer Schleifensteuereinheit **508** gesteuert wird. Die Schleife **506** weist einen Geber **510** auf, der in Bezug auf die Steuerventile **504** und **506** distal von der Schleifensteuerung **508** positioniert ist.

[0076] Typischerweise ist das primäre Prozeßsteuerventil **502** aktiv, und das redundante Prozeßsteuerventil **504** ist im Standby- oder Bypassstatus. Die Schleife **506** nutzt die digitale Zweiwege-Kommunikation, so daß die Steuerventile **504** und **506** und der Geber **510** sämtlich Meldungen empfangen und Meldungen an die Schleifensteuereinheit **508** senden. Daher empfängt die Schleifensteuerung **508** Information, die den präzisen Status von funktionalen Elementen innerhalb des Steuersystemnetzes **500** bezeichnet.

[0077] Bei Empfang von Information, die einen Ausfall oder anderen fehlerhaften Status eines funktionalen Elements bedeutet, löst die Schleifensteuereinheit **508** eine Antwort aus, um das ausfallende funktionale Element zu deaktivieren und ein redundantes Element, falls verfügbar, zu aktivieren. Die Schleifensteuereinheit **508** deaktiviert typischerweise ausgewählte funktionale Elemente, indem sie die funktionalen Elemente in eine selbstschützende Betriebsart bringt.

[0078] Bei einigen Ausführungsformen sind die Steuerventile **504** und **506** so eingestellt, daß sie im Bypassmodus mit halber Leistung wirksam sind, und die Reaktion auf einen Ausfall eines einzelnen Ventils ist die Deaktivierung des ausfallenden Ventils und die Aktivierung des funktionalen Ventils auf volle Kapazität. Ferner können die Ventile **504** und **506** in Reihe geschaltet sein, so daß das eine geöffnet bleibt, während das andere den Durchfluß steuert.

[0079] Unter Bezugnahme auf [Fig. 10](#) zeigt ein schematisches Blockbild ein Steuersystemnetz **600**, das zwei funktionale Elemente aufweist, und zwar einen primären Geber **602** und einen redundanten Geber **604**, die in einen Prozeßdurchflußstrom **612** eingekoppelt sind. In dem Steuersystemnetz **600** sind der primäre Geber **602** und der redundante Geber **604** mit einer einzelnen Zweidrahtschleife **606** gekoppelt, die von einer Schleifensteuereinheit **608** ge-

steuert wird. Der primäre Geber **602** ist aktiv, und der redundante Geber **604** ist im Standby- oder Bypaßstatus. Die Schleife **606** nutzt die digitale Zweige-Kommunikation, so daß beide Geber **604** und **606** Meldungen empfangen und Meldungen an die Schleifensteuereinheit **608** senden. Somit empfängt die Schleifensteuereinheit **608** Information, die den präzisen Status von funktionalen Elementen innerhalb des Steuersystemnetzes **600** bezeichnet.

**[0080]** Bei Empfang von Information, die einen Ausfall oder sonstigen fehlerhaften Status eines Gebers bezeichnet, ignoriert die Schleifensteuereinheit **608** einfach Übertragungen von einem nicht-funktionalen Geber.

**[0081]** [Fig. 11](#) zeigt ein Blockbild einer Prozeßsteuerschleife **700**, die verteilte Steuerfunktionen hat, die implementiert sind unter Verwendung von redundanten Funktionsblöcken wie etwa denen in einem Fieldbus-Kommunikationsnetz. Die Schleife **700**, die eine einfache Rückführungs-Ventilsteuerschleife implementieren kann, wie sie [Fig. 4](#) zugeordnet ist, weist hier einen einzelnen AI-Funktionsblock **702** auf, der mit einem Paar von redundanten PID-Funktionsblöcken **704** und **706** gekoppelt ist, die ihrerseits durch einen Fehlerdetektions-Funktionsblock **708** mit einem Paar von redundanten AO-Funktionsblöcken **710** und **712** gekoppelt sind. Im Betrieb wirken der AI-Funktionsblock **702**, einer der PID-Funktionsblöcke **704** oder **706** und einer der AO-Funktionsblöcke **710** und **712** mit dem Fehlerdetektions-Funktionsblock **708** zusammen, um die einfache Rückführungssteuerschleife zu implementieren. Wie aus [Fig. 11](#) ersichtlich ist, leitet der AI-Funktionsblock **702** seinen Ausgang an die PID-Funktionsblöcke **704** und **706**, von denen einer wirksam ist, um ein Steuer-signal zu erzeugen, das durch den Fehlerdetektions-Funktionsblock **708** zu einem der AO-Funktionsblöcke **710** oder **712** geleitet wird. Derselbe PID-Funktionsblock **704** oder **706** empfängt auch ein Rückkopplungssignal von dem einen von den AO-Funktionsblöcken **710** oder **712** durch den Fehlerdetektions-Funktionsblock **708** über eine von Rückkopplungsleitungen **714** oder **716**. So kann beispielsweise im Normalbetrieb die Schleife **700** so wirksam sein, daß der PID-Funktionsblock **704** und der AO-Funktionsblock **710** durch den Fehlerdetektions-Funktionsblock **708** gekoppelt sind. Der Fehlerdetektions-Funktionsblock **708** analysiert den Modus der Blöcke **704** und **710** (sowie der Blöcke **706** und **712**) oder analysiert die von den Blöcken **704** und **710** empfangenen Signale, um zu detektieren, ob einer dieser Funktionsblöcke fehlerhaft arbeitet. Wenn der Fehlerdetektions-Funktionsblock **708** einen Fehlerstatus in einem der Blöcke **704** oder **710** detektiert, veranlaßt der Fehlerdetektionsblock **708** sofort einen redundanten Funktionsblock, und zwar entweder den redundanten PID-Funktionsblock **706** (falls der PID-Funktionsblock **704** fehlerhaft ist) oder den red-

undanten AO-Funktionsblock **712** (falls der AO-Funktionsblock **710** fehlerhaft ist), in der Schleife **700** wirksam zu werden, um dadurch den fehlerhaften Funktionsblock aus der Schleife **700** zu entfernen, was wiederum der Schleife **700** erlaubt, ihre Operation weiter auszuführen, ohne daß die Schleife **700** oder der Prozeß, in den die Schleife **700** eingekoppelt ist, abgeschaltet werden muß.

**[0082]** Selbstverständlich kann der Fehlerdetektions-Funktionsblock **708** die Operation der Schleife **700** auf jede gewünschte Weise schalten, was einschließt, daß die beiden redundanten Funktionsblöcke **706** und **712** so geschaltet werden, daß sie zusammenwirken, wenn einer der Funktionsblöcke **704** oder **710** fehlerhaft ist, daß die Schleife **700** so geschaltet wird, daß der PID-Funktionsblock **704** und der AO-Funktionsblock **712** zusammenwirken, wenn z. B. der AO-Funktionsblock **710** fehlerhaft ist, oder daß die Schleife **700** so geschaltet wird, daß der redundante PID-Funktionsblock **706** und der AO-Funktionsblock **710** zusammenwirken, wenn z. B. der PID-Funktionsblock **704** ausfällt. Ebenso kann der Fehlerdetektions-Funktionsblock **708** zwischen jede Gruppe von redundanten Funktionsblöcken und einen einzelnen Funktionsblock oder zwischen jeweils zwei Gruppen von redundanten Funktionsblöcken innerhalb einer Prozeßsteuerschleife eingekoppelt werden, um dort die Redundanz vorzusehen. Außerdem kann die Redundanz erreicht werden, indem mindestens ein redundanter Funktionsblock für jeden der Funktionsblöcke innerhalb einer Schleife vorgesehen wird, etwa durch Einfügen eines redundanten AI-Funktionsblocks in die Schleife **700** von [Fig. 11](#). Weniger Redundanz kann jedoch erreicht werden, indem ein redundanter Funktionsblock für nur einen Funktionsblock innerhalb einer Schleife oder nur für eine begrenzte Anzahl der Funktionsblöcke innerhalb einer Schleife vorgesehen wird. Es versteht sich außerdem, daß der Fehlerdetektions-Funktionsblock **708** auf jede gewünschte Weise eingekoppelt sein kann und in jedem Gerät innerhalb eines Prozeßsteuerungssystems positioniert sein kann, solange der Fehlerdetektions-Funktionsblock **708** kommunikativ mit den anderen Funktionsblöcken innerhalb einer redundanten Schleife über einen Bus wie etwa einen Fieldbus-Kommunikationsbus verbunden ist. Auch können die redundanten Funktionsblöcke, z. B. die Blöcke **704** und **706** oder die Blöcke **710** und **712**, in denselben oder in verschiedenen Geräten vorgesehen sein.

**[0083]** Falls gewünscht, können außerdem die Ausgänge der PID-Funktionsblöcke **704** und **706** direkt mit den AO-Funktionsblöcken **710** und **712** (sowie mit dem Fehlerdetektions-Funktionsblock **708**) in [Fig. 11](#) gekoppelt sein, während die Rückführung von den AO-Funktionsblöcken **710** und **712** direkt mit den PID-Funktionsblöcken **704** und **706** (sowie dem Fehlerdetektions-Funktionsblock **708**) gekoppelt sein

kann. Bei dieser Konfiguration detektiert der Fehlerdetektions-Funktionsblock **708** Fehler innerhalb der Funktionsblöcke **704** und **706** oder **710** und **712** und bewirkt, daß ein fehlerhafter Funktionsblock aus der Schleife ausgekoppelt wird und gleichzeitig der zugehörige redundante Funktionsblock in die Schleife eingekoppelt wird, ohne daß tatsächlich irgendwelche Signale beispielsweise zwischen den PID- und den AO-Funktionsblöcken innerhalb der Schleife **700** gesendet werden.

**[0084]** [Fig. 12](#) ist ein schematisches Blockbild eines Steuersystemnetzes **100**, das eine Feldgerät-Redundanz implementiert, indem beispielsweise irgendwelche oder sämtliche der redundanten Verbindungen, die in den [Fig. 6](#) bis [Fig. 11](#) gezeigt sind, sowie irgendwelche anderen redundanten Verbindungen genutzt werden. Das gezeigte Steuersystemnetz **100** weist einen Computer **102** wie etwa einen Personalcomputer oder einen Arbeitsplatzrechner auf, der mit einem Netzbus **104** über eine Steuereinheit **106** wie etwa ein digitales Steuersystem und ein Paar von redundanten Kommunikationsleitungen **107** gekoppelt ist. Der Netzbus **104** weist eine primäre Schleife **112** und eine redundante Schleife **113** auf, die jeweils schleifengespeiste digitale Zweidraht-Zweiwege-Kommunikationen beispielsweise entsprechend dem Feldbus-Protokoll oder einem anderen Kommunikationsprotokoll, das einem Prozeßsteuerungssystem mit verteilten Steuerfunktionen zugeordnet ist, implementieren. Das Steuersystemnetz **100** kommuniziert mit einem externen Netz **114** über eine Kopplung des Netzbusses **104** an einem Knotenpunkt **115**. Das Steuersystemnetz **100** weist eine Vielzahl von Feldgeräten **116** auf, die entweder direkt in den Netzbus **104** eingekoppelt sind oder mit dem Netzbus **104** über Brücken **118** und lokale Busse **120** gekoppelt sind. Bei dem gezeigten Steuersystemnetz **100** ist ein lokaler Bus **120** (mit **122** bezeichnet) mit dem Knotenpunkt **115** über einen externen netzredundanten Bus **124** gekoppelt, der eine primäre Schleife **126** und eine redundante Schleife **128** hat.

**[0085]** Eine Redundanz kann selektiv auf Feldgeräteebene durch ein primäres Feldgerät (mit **130** bezeichnet) und ein redundantes Feldgerät (mit **132** bezeichnet) implementiert werden, die mit einer ersten Brücke (mit **134** bezeichnet) über eine redundante Verbindung **136** mit einem lokalen Bus **138** verbunden sind und mit einer zweiten Brücke (mit **140** bezeichnet) über eine redundante Verbindung **142** mit dem lokalen Bus **122** verbunden sind. Ein vollständig redundantes funktionales Element hat die gleiche Funktionsmengen- oder Funktionsblockfähigkeit wie ein entsprechendes primäres funktionales Element. Ein funktionales Element mit begrenzter Redundanz hat eine Funktionsmenge, die mindestens eine Funktion oder ein Merkmal eines entsprechenden primären funktionalen Elements nicht hat.

**[0086]** Das gezeigte Steuersystemnetz **100** implementiert die Redundanz auf vielen Ebenen in einer digitalen, schleifengespeisten Zweidraht-Zweiwege-Kommunikationsumgebung, in einer Vierdraht-Kommunikationsumgebung oder jeder anderen Prozeßsteuerungsumgebung, die verteilte Steuerfunktionen anwendet. Erstens ist der Computer **102** mit der Steuereinheit **106** unter Verwendung redundanter Leitungen **107** verbunden. Zweitens weist der Netzbus **104** eine primäre Schleife oder einen primären Bus **112** und eine redundante Schleife oder einen redundanten Bus **113** auf. Drittens sind die Brücken **118** und direkt angeschlossene Feldgeräte **116** mit dem Netzbus **104** mit redundanten Verbindungen gekoppelt. Viertens sind das primäre Feldgerät **130** und das redundante Feldgerät **132** über die redundante Verbindung **136** mit dem lokalen Bus **138** mit der ersten Brücke **134** gekoppelt. Fünftens sind das primäre Feldgerät **130** und das redundante Feldgerät **132** mit der ersten Brücke **134** gekoppelt über eine redundante Verbindung **136** mit einem lokalen Bus **138** und sind mit einer zweiten Brücke **140** gekoppelt über eine redundante Verbindung **142** mit dem lokalen Bus **122**. Sechstens ist der lokale Bus **122** mit dem externen Netz **114** an dem Knotenpunkt **115** durch den Netzbus **104** und den redundanten Bus **124** des externen Netzes redundant gekoppelt. Siebtens ist der redundante Bus **124** des externen Netzes ein redundanter Bus. Achters sind redundante Funktionsblöcke in den Geräten (beispielsweise den Geräten **116**) angeordnet, die mit dem Netz **100** gekoppelt sind.

**[0087]** Bei anderen Ausführungsformen eines Steuersystemnetzes wird die Redundanz selektiv für den Netzbus **104** alleine implementiert oder wird für ausgewählte Feldgeräte **116**, sämtliche Feldgeräte **116** oder keine Feldgeräte **116** implementiert. Ebenso ist eine Redundanz der Verbindungen des lokalen Busses **120** mit dem Knotenpunkt **115** und von Funktionsblöcken fakultativ.

**[0088]** Das Steuersystemnetz **100**, das die Feldgeräte-Redundanz implementiert, ist wirksam für Schleifen, die schleifengespeiste digitale Zweidraht-Zweiwege-Kommunikationen implementieren, sowie für Vierdraht-Schleifen oder andere Schleifen, die Prozeßsteuerfunktionen auf eine verteilte Weise implementieren, was Schleifen einschließt, die einen Feldbus-Standard (Feldbus Foundation, Austin, Texas), einen WORLDFIP-Standard, einen LONWORKS-Standard, einen PROFIBUS-Standard, einen anderen SP-50-Kommunikationsstandard und dergleichen implementieren. Das Steuersystemnetz **100**, das die Feldgeräte-Redundanz implementiert, ist außerdem für Schleifen wirksam, die gemischte Analog-/Digital-Protokolle implementieren, was beispielsweise den HART-Standard einschließt.

**[0089]** [Fig. 13](#) zeigt als schematisches Blockbild ei-

nes der digitalen Feldgeräte **116** (von [Fig. 12](#)), das eine schleifengespeiste, digital kommunizierende Zweidraht-Zweiwege-Kombination aus Positionierer/Ventil ist. Das digitale Feldgerät **116** weist auf: eine Feldgerät-Steuereinheit **1102**, einen I/P-Wandler **1104**, ein pneumatisches Relais **1106**, einen Betätiger **1108** und ein Ventil **1109**, die über verschiedene Pneumatik- und elektrische Leitungen miteinander gekoppelt sind.

**[0090]** Das Feldgerät **116** empfängt Betriebssignale und überträgt Statusinformation und Daten in digitaler Form auf dem Zweidrahtbus **122**, bevorzugt nach dem Fieldbus-Standard, und ist daher ein Zweidraht-Positionierer. Gleichmaßen empfängt das Feldgerät **116** Energie, hauptsächlich zum Treiben der Gerätesteuereinheit **1102** und des I/P-Wandlers **1104**, über das eine kontinuierliche Schleife bildende Zweidraht-Busselement **120** und ist daher ein schleifengespeistes Gerät.

**[0091]** Wie [Fig. 13](#) zeigt, ist der I/P-Wandler **1104** mit der Gerätesteuereinheit **1102** über eine I/P-Wandlersteuerleitung **1110** elektrisch gekoppelt und kommuniziert bei der gezeigten Ausführungsform mit der Gerätesteuereinheit **1102** unter Nutzung von analogen Steuersignalen.

**[0092]** Der I/P-Wandler **1104** erzeugt ein pneumatisches Signal, das die Betätigung des Ventils **1109** auslöst und in elektromechanischen Geräten zur Umwandlung elektrischer Signale in Luftdruck für einen pneumatischen Positionierer außerordentlich nützlich ist. Der Betätiger **1108** steuert die Position eines Ventilelements **1114** (das eine Ventilspindel sein kann) des Ventils **1109**, während ein Positionsgeber **1116** die Position des Ventilelements **1114** erfaßt und ein Rückführungssignal erzeugt, das auf einer Signalleitung **1117** an die Gerätesteuereinheit **1102** übermittelt wird. Dieses Positionssignal kann von der Gerätesteuereinheit **1102** genutzt werden, um den Betrieb des Feldgeräts **116** so zu steuern, daß der I/P-Wandler **1104** den Luftdruck auf eine Weise steuert, die bewirkt, daß das Ventilelement **1114** sich an einer gewünschten Position befindet. Positions- und andere Rückführungsinformation kann in einer Speichereinheit oder einem Speicher der Gerätesteuereinheit **1102** gespeichert werden, und Zugriff darauf kann von außen über den Bus **120** erfolgen, um z. B. einen Fehlerstatus des Geräts **116** zu detektieren.

**[0093]** Gemäß dem Standard empfängt das Feldgerät **116** Druckluft von einer externen Quelle (nicht gezeigt) über eine Druckluftleitung **1118**, die mit dem I/P-Wandler **1104** und dem pneumatischen Relais **1106** verbunden ist. Ein Eingangssensor **1120**, der typischerweise zwischen der externen Druckluftquelle und dem I/P-Wandler **1104** positioniert ist, mißt den eingehenden Speiseluftdruck in der Pneumatikleitung **1118** und gibt diesen Meßwert an die Geräte-

steuereinheit **1102** weiter. Der I/P-Wandler **1104** ist mit dem pneumatischen Relais **1106** über eine pneumatische Steuerleitung **1122** verbunden, und ein I/P-Sensor **1124** ist zwischen dem I/P-Wandler **1104** und dem pneumatischen Relais **1106** positioniert, um den Speiseluftdruck in der Leitung **1122** zu messen. Ebenso ist das pneumatische Relais **1106** mit dem Betätiger **1108** über eine pneumatische Betätigungsleitung **1126** verbunden, und ein Relaisensor **1128** ist zwischen dem pneumatischen Relais **1106** und dem Betätiger **1108** positioniert, um den Speiseluftdruck in der Leitung **1126** zu messen. Die Pneumatikleitungen **1118**, **1122** und **1126** werden als Teile einer einzigen Pneumatikleitung betrachtet, die den Wandler **1104** mit dem Ventil **1109** koppelt.

**[0094]** Im Betrieb steuert die Gerätesteuereinheit **1102** die Betätigung des Ventils **1109** durch Steuerung des I/P-Wandlers **1104**, um einen gesteuerten Ventilbetriebsdruck in der pneumatischen Steuerleitung **1126** einzustellen. Die Gerätesteuereinheit **1102** sendet ein Steuersignal auf der I/P-Wandlersteuerleitung **1110** an den I/P-Wandler **1104**, um einen Ausgangsdruck der Kombination aus I/P-Wandler **1104** und Relais **1106** so zu steuern, daß er zwischen ungefähr 0,21 bis 7,06 kscm (3 bis 100 psi) liegt, der einem Steuereingang des Betätigers **1108** zugeführt wird. Der Betätiger **1108** erzeugt einen Ausgangsdruck, der zur Betätigung des Ventils **1109** angelegt wird.

**[0095]** Wie bekannt ist, wandelt also der I/P-Wandler **1104** elektrische Signale in ein pneumatisches Luftdrucksignal um. Ein Beispiel eines geeigneten I/P-Wandlers **1104** ist in den US-PS 5 439 021 mit dem Titel "Electro-Pneumatic Converter", erteilt für B. J. Burlage et al. am 8. August 1995, beschrieben; diese Druckschrift wird hier summarisch eingeführt. Ebenso wird das pneumatische Relais **1106**, das als pneumatischer Verstärker dient, von dem I/P-Wandler **1104** entsprechend den Vorgaben der Gerätesteuereinheit **1102** so gesteuert, daß der Luftdruck der pneumatischen Betätigungssignalleitung **1126** auf einen gesteuerten Wert erhöht wird. Allgemein gesagt, liefert also das pneumatische Relais **1106** einen gesteuerten Ausgangsdruck an eine Last oder ein Gebrauchsgerät wie etwa einen Betätigen oder einen Kolben in Abhängigkeit von einem Steuersignal von der Gerätesteuereinheit **1102**. Ein geeignetes Relais ist in der US-PS 4 974 625 mit dem Titel "Four Mode Pneumatic Relais", erteilt für S. B. Paullus et al. am 4. Dezember 1990, beschrieben; dieses Dokument wird hier summarisch eingeführt. Bei den gezeigten Ausführungsform ist das Relais **1106** ein multifunktionales Vier-Moden-Pneumatikrelais, das für jede Betriebskombination von direkt/schnapp, direkt/proportional, invertiert/schnapp oder invertiert/proportional konfigurierbar ist. Im Proportionalmodus entwickelt das pneumatische Relais **1106** einen Druckausgang, den zu einem eingegebenen Druck

bzw. einer eingegebenen Kraft proportional ist. In einem Ein-/Aus- oder Schnappmodus erzeugt das pneumatische Relais **1106** einen konstanten Druckausgang, der gewöhnlich gleich dem Druck des zugeführten Speisedrucks ist, in Abhängigkeit von dem Aufbringen eines definierten Bereichs von Kraft- oder Druck-Steuereingängen. In einem direkten Betriebsmodus steigt der Ausgangsdruck des pneumatischen Relais **106** mit einem ansteigenden Eingangssignal. In einem invertierten Betriebsmodus nimmt der Ausgangsdruck des Relais mit einem ansteigenden Eingangssignal ab.

**[0096]** Der Eingangssensor **1120**, den I/P-Sensor **1124** und der Relaisensor **1128** sind Druckwandler, die einen Umwandler zum Umwandeln eines Drucksignals in ein elektrisches Signal enthalten und Rückführungssignale an die Gerätesteuerung **1102** auf einer Leitung **1130** liefern. Den I/P-Sensor **1124** ist diagnostisch nützlich, um eine Störung entweder des I/P-Wandlers **1104** oder des pneumatischen Relais **1106** zu detektieren und beispielsweise zu bestimmen, ob ein Ausfall ein mechanischer Ausfall oder ein elektrischer Ausfall ist. Der I/P-Sensor **1124** ist außerdem nützlich zum Detektieren von einigen Systemproblemen einschließlich einer Bestimmung, ob der in das digitale Feldgerät **16** eingegebene Luftdruck ausreichend ist. Somit ermöglicht der I/P-Sensor **1124** eine rasche Diagnose des Status des I/P-Wandlers **1104** und des pneumatischen Relais **1106**, so daß diese Einrichtungen erforderlichenfalls schnell ausgewechselt werden können und eine Prozeßsteuereinheit aktiviert werden kann, um auf den Gebrauch eines anderen redundanten Geräts umzuschalten, wenn dies möglich ist.

**[0097]** Bei einer Ausführungsform ist ein geeignetes Ventil **1109** zur Verwendung in dem digitalen Feldgerät **116** eine Ventil- und Betätigeranordnung, die einen Feder-/Membran-Betätiger an einem Kolbenlängsschieberventil verwendet, das in einem Analoggerät verwendet wird, das in der US-PS 4 976 144 mit dem Titel "Diagnostic Apparatus and Method for Fluid Control Valves" beschrieben wird, erteilt am 11. Dez. 1990 für W. V. Fitzgerald; dieses Dokument wird hier summarisch eingeführt. Bei dieser beispielhaften Ausführungsform wird dem Betätiger **108** ein Drucksignal von ungefähr 0,21 kscm (3 psi) zugeführt in Abhängigkeit von einem ungefähren 4-mA-Signal, das von der Gerätesteueereinheit **1102** an den I/P-Wandler **1104** angelegt wird, was in einem entsprechenden Druck in der pneumatischen Betätigungssignalleitung **1126** resultiert, der ausreicht, um das Ventil **1109** aus einer vollständig geöffneten Position zu verschieben. Wenn die Feldgerätsteuereinheit **1102** den an den I/P-Wandler **104** angelegten Steuerstrom zu ungefähr 20 mA ändert, erzeugt der I/P-Wandler **1104** einen Druck in der pneumatischen Betätigungsleitung **1126** von ungefähr 1,06 kscm (15 psi), wodurch das Ventil **1109** in eine vollständig ge-

schlossene Position gedrückt wird. Verschiedene Positionen des Ventils **1109** zwischen der vollständig geöffneten und der vollständig geschlossenen Position werden erreicht durch den Betrieb der Gerätesteueereinheit **1102**, die den an den I/P-Wandler **1104** angelegten Eingangsstrom in dem Bereich zwischen 4 mA und 20 mA steuert.

**[0098]** Die Gerätesteuerung **1102** führt digitale Übertragungen mit relativ hoher Geschwindigkeit aus, um Steuersignale zu empfangen und Positions- und Druckinformation an einen externen Prozessor oder Arbeitsplatzrechner in dem Prozeßsteuerungsnetz über den Bus **120** zu senden. Die Gerätesteueereinheit **1102** weist einen Speicher auf, um die Ergebnisse einer Vielzahl von Diagnosetests zu speichern, so daß relevante Daten für die Analyse verfügbar sind. Diagnoseoperationen wie etwa Gerätediagnosen liegen allgemein in Form von Softwareprogrammcodes vor und werden typischerweise in der Gerätesteueereinheit **1102** des Feldgeräts **116** verschlüsselt, gespeichert und ausgeführt.

**[0099]** Eine Gerätediagnose-Bewertung des Ventils **1109** kann durch die Operation der Gerätesteueereinheit **1102** zur Steuerung des an den I/P-Wandler **1104** angelegten Eingangsstroms in einem Bereich durchgeführt werden, der ausreicht, um das Ventil **119** zwischen der vollständig geöffneten und der vollständig geschlossenen Position zu testen. Während der Gerätediagnose-Bewertung werden die Ausgänge des Eingangssensors **1120**, des I/P-Sensors **1124** und des Relaisensors **1128** von der Gerätesteueereinheit **1102** überwacht, um den pneumatischen Druck in den Pneumatikleitungen **1118**, **1122** und **1126** zu erfassen, die für die Analyse genutzt werden. Der Ausgang des Positionssensors **1116** wird ebenfalls überwacht, um die Position oder Bewegung des Ventilkolbens **1114** zu detektieren, die einer Position oder Bewegung des Ventilschiebers (nicht gezeigt) in dem Ventil **1109** entspricht.

**[0100]** Ein Testbetriebszyklus des Ventils **1109** wird also unter Steuerung durch die Gerätesteueereinheit **1102** durchgeführt, indem ein gesteuerter variabler Druck an den I/P-Wandler **1104** angelegt, der Druck in den Pneumatikleitungen **1118**, **1122** und **1126** erfaßt und die Position des Ventilkolbens **1114** unter Anwendung des Positionssensors **1116** erfaßt wird. Auf diese Weise empfängt die Gerätesteueereinheit **1102** gleichzeitig zeitlich veränderliche elektrische Signale, welche die Drücke an den beispielhaften Stellen und die Position des Ventils **1109** bezeichnen, und kann diese Signale dazu nutzen, alle möglichen Gerätediagnose-Parameter auf jede bekannte oder gewünschte Weise zu bestimmen.

**[0101]** Bei einer Ausführungsform werden der I/P-Wandler **1104** und das pneumatische Relais **1106** getestet unter Anwendung eines Diagnosetestver-

fahrens, bei dem der I/P-Wandler **1104** in den vollständig geöffneten Zustand getrieben wird, um den vollen Luftdruck zu messen, der dem Ventil **1109** zugeführt wird. Während der I/P-Wandler **1104** in den geöffneten Zustand getrieben wird, mißt der I/P-Sensor **1124** ständig den Druck in der pneumatischen Steuerleitung **1122**. Wenn der Druck niedriger zu werden beginnt, zeigt der Test, daß die Luftzufuhr unzureichend sein kann. Ein weiterer Diagnosetest der ausreichenden Luftzufuhr wird durchgeführt durch Pumpen des Ventils **1109** durch Anlegen eines oszillierenden Signals an den I/P-Wandler **1104**, so daß das Ventil **1109** einen Saugvorgang in Bezug auf die Luftzufuhr beginnt, wonach der maximale Durchfluß und die maximalen Druckwerte unter Anwendung des I/P-Sensors **1124** gemessen werden.

[0102] Wie [Fig. 14](#) zeigt, umfaßt die Gerätesteuereinheit **1102** einen Mikroprozessor **1140**, eine Schnittstelle **1142**, eine Bustrennschaltung **1144**, eine Vielzahl von Speichereinrichtungen wie etwa einen Direktzugriffsspeicher bzw. RAM **1146**, einen Lesespeicher bzw. ROM **1148** und einen nichtflüchtigen Direktzugriffsspeicher bzw. NVRAM **1150**, und eine Vielzahl von Signalverarbeitungseinrichtungen wie einen A/D-Wandler **1152**, einen D/A-Wandler **1154** und einen Multiplexer **1156**. Die Schnittstelle **1142** (die ein Busverbinder ist) ist eine Schaltung, die eine Seriell-Parallel-Protokollumwandlung und eine Parallel-Seriell-Protokollumwandlung ausführt und dazu genutzt wird, Rahmeninformation zu Datenpaketen entsprechend jeder gewünschten Protokolldefinition wie etwa dem Feldbus-Protokoll hinzuzufügen. Die Bustrennschaltung **1144** ist eine Schaltung, die dazu genutzt wird, ein Zweidrahtmedium-Kommunikationssignal auf dem Bus **120** in eine digitale Darstellung des Kommunikationssignals umzuwandeln, und liefert von dem Bus **120** empfangene Energie an andere Schaltungen in der Gerätesteuereinheit **1102** sowie an den I/P-Wandler **1104**. Die Bustrennschaltung **1144** kann außerdem eine Wellenformung und Signalgabe auf dem Bus **120** ausführen.

[0103] Der A/D-Wandler **1152** ist mit Wandlern wie etwa den Positions- und Druckwandlern des Positionssensors **1116** und der Drucksensoren **1120**, **1124** und **1128** von [Fig. 13](#) sowie mit anderen gewünschten Analogeingangseinrichtungen gekoppelt. Der A/D-Wandler **1152** kann zwar eine begrenzte Anzahl von Eingangskanälen haben, aber der Multiplexer **1156** kann genutzt werden, um die Abtastung einer Vielzahl von Signalen zu ermöglichen. Falls gewünscht, kann der Multiplexer **1156** eine Reihe von Verstärkern aufweisen, die zwischen die Signalleitungen **1117** und **1130** ([Fig. 13](#)) gekoppelt sind, um die zugeführten Positions-, Druck- und anderen Rückführungssignale zu verstärken. Der D/A-Wandler **1154** führt die Digital-/Analog-Umwandlung an Signalen aus, die von dem Mikroprozessor **1140** erzeugt werden, um analogen Komponenten wie etwa dem

I/P-Wandler **1104** zugeführt zu werden.

[0104] Die gezeigten Ausführungsformen eines Steuersystemnetzes zur Implementierung einer Redundanz sehen vorteilhafterweise Sicherheit für eine Schleife durch Implementierung von schleifengepeisten digitalen Zweidraht-Zweiwege-Kommunikationen oder anderen Kommunikationen vor durch Beibehaltung der Operation des Steuersystemnetzes ungeachtet des Ausfalls eines funktionalen Elements. Dieser Vorteil ist in Prozeßsteuersystemen wichtig, bei denen die Kosten der Abschaltung einer Prozeßsteuerleitung enorm sind.

[0105] Selbstverständlich kann das Prozeßsteuerungsnetz, das redundante Elemente hat, eine Redundanz in anderen Konfigurationen nach Wunsch anwenden. Außerdem wurde zwar hier beschrieben, daß das Prozeßsteuerungsnetz mit redundanten Elementen Geber und Positionierer-/Ventileinrichtungen aufweist, es ist aber zu beachten, daß dieses Netz andere Arten von Geräten aufweisen kann, beispielsweise solche mit bewegbaren Elementen wie etwa Registern, Gebläsen usw. sowie Steuereinheiten, Brückeneinrichtungen, Sensoren usw.

[0106] Ferner wird zwar die Schaltlogik des Prozeßsteuerungsnetzes mit redundanten Elementen gemäß der vorliegenden Beschreibung bevorzugt in Software implementiert, die beispielsweise in einer Prozeßsteuerungseinrichtung oder einer Steuereinheit gespeichert ist, sie kann aber alternativ oder zusätzlich nach Wunsch in Hardware, Firmware usw. implementiert werden. Wenn sie in Software implementiert ist, kann diese Logik in jedem computerlesbaren Speicher wie etwa auf einer Magnetplatte, einer Laserplatte oder einem anderen Speichermedium, in einem RAM oder ROM eines Computers usw. gespeichert sein. Ebenso kann diese Software einem Anwender oder einer Einrichtung mittels jeder bekannten oder gewünschten Liefermethode geliefert werden, beispielsweise über einen Kommunikationskanal wie eine Telefonleitung, das Internet usw.

[0107] Vorstehend werden zwar verschiedene Merkmale und Ausführungsformen der Erfindung beschrieben, diese können aber ohne weiteres miteinander kombiniert werden und in weiteren Ausführungsformen der Erfindung resultieren.

### Patentansprüche

1. Prozeßsteuerungssystem (**100**, **200**, **300**, **400**, **500**, **600**, **700**), das auf verteilte Weise Prozeßsteuerungsfunktionen innerhalb eines Prozesses ausführt und folgendes aufweist:  
einen Kommunikationsbus (**202**), der eine Kommunikationsprozessfunktion in dem Prozess ausführt;  
eine Vielzahl von Feldgeräten (**204**), die über den Kommunikationsbus (**202**) kommunikativ gekoppelt

sind, wobei jedes der Feldgeräte (204) eine andere Prozessfunktion innerhalb des Prozesses ausführt; gekennzeichnet durch eine Steuereinheit (208), die mit dem Kommunikationsbus (202) gekoppelt ist, wobei die Steuereinheit (208) über den Kommunikationsbus (202) mit der Vielzahl von Feldgeräten (204) kommunikativ gekoppelt ist und wobei die Steuereinheit (208) über den Kommunikationsbus (202) die Operation der Vielzahl von Feldgeräten (204) überwacht; und ein Paar von redundanten Elementen, die ein erstes redundantes Element (204) und ein zweites redundantes Element (206) aufweisen, die ausgebildet sind, um die gleiche Prozessfunktion innerhalb des Prozesses auszuführen; wobei die Steuereinheit (208) über den Kommunikationsbus (202) mit dem Paar von redundanten Elementen (204, 206) kommunikativ gekoppelt ist, um unter Nutzung des Kommunikationsbusses (202) einen Ausfall von einem von den redundanten Elementen zu detektieren und um beim Detektieren des Ausfalls des einen von den redundanten Elementen das andere von den redundanten Elementen mit dem Prozesssteuerungssystem funktionsmäßig zu verbinden.

2. Prozesssteuerungssystem (200) nach Anspruch 1, wobei der Kommunikationsbus (202) ein schleifengespeistes, digitales Zweidraht-Zweiwege-Kommunikationsprotokoll implementiert.

3. Prozesssteuerungssystem (200) nach Anspruch 2, wobei das Kommunikationsprotokoll ein Fieldbus-Kommunikationsprotokoll ist.

4. Prozesssteuerungssystem (200) nach Anspruch 1, wobei der Kommunikationsbus (202) ein Vierdraht-Kommunikationsprotokoll implementiert.

5. Prozesssteuerungssystem (200) nach Anspruch 1, wobei der Kommunikationsbus (202) ein schleifengespeistes, gemischt digitales und analoges Zweidraht-Zweiwege-Kommunikationsprotokoll implementiert.

6. Prozesssteuerungssystem (300, 400) nach Anspruch 1, wobei der Kommunikationsbus einen ersten redundanten Kommunikationsbus (302, 402) und einen zweiten redundanten Kommunikationsbus (303, 403) aufweist, und wobei das erste redundante Element den ersten redundanten Kommunikationsbus (302, 402) aufweist und das zweite redundante Element den zweiten redundanten Kommunikationsbus (303, 403) aufweist.

7. Prozesssteuerungssystem (500, 600) nach Anspruch 1, wobei das erste redundante Element eines von der Vielzahl von Feldgeräten (502, 602) aufweist und das zweite redundante Element ein weiteres Feldgerät (504, 604) aufweist, das mit dem Kom-

munikationsbus (506, 606) gekoppelt ist.

8. Prozesssteuerungssystem (500) nach Anspruch 7, wobei das erste redundante Element und das zweite redundante Element Ventile (502, 504) sind, die funktionsmäßig parallel zueinander in den Prozess eingekoppelt sind.

9. Prozesssteuerungssystem (600) nach Anspruch 7, wobei das erste redundante Element und das zweite redundante Element Geber (602, 604) sind, die funktionsmäßig in Reihe miteinander in den Prozess eingekoppelt sind.

10. Prozesssteuerungssystem (700) nach Anspruch 1, wobei das erste redundante Element einen ersten Funktionsblock (704) aufweist, der eine bestimmte Prozessfunktion ausführt, und das zweite redundante Element einen zweiten Funktionsblock (706) aufweist, der die bestimmte Prozessfunktion ausführt.

11. Prozesssteuerungssystem (700) nach Anspruch 10, wobei der erste (704) und der zweite Funktionsblock (706) in verschiedenen von der Vielzahl von Feldgeräten vorliegen.

12. Prozesssteuerungssystem (700) nach Anspruch 10, wobei die Steuereinheit (702) einen mit dem ersten (704) und dem zweiten Funktionsblock (706) kommunikativ gekoppelten weiteren Funktionsblock (708) aufweist, der eine Funktionsstörung von einem von dem ersten (704) und dem zweiten Funktionsblock (706) detektiert.

13. Prozesssteuerungssystem (300) nach Anspruch 1, wobei der Kommunikationsbus einen ersten redundanten Kommunikationsbus (302) und einen zweiten redundanten Kommunikationsbus (303) aufweist, wobei das erste redundante Element eine Schleife unter Einschluss des ersten redundanten Kommunikationsbusses (302) in Verbindung mit einem von den Feldgeräten (304) aufweist, und wobei das zweite redundante Element den zweiten redundanten Kommunikationsbus (303) in Verbindung mit einem redundanten Feldgerät (306) aufweist.

14. Prozesssteuerungssystem (200) nach Anspruch 1, das ferner eine Steuerlogik aufweist, die in einem dem Paar von redundanten Elementen (204, 206) zugeordneten funktionalen Element wirksam ist, wobei die Steuerlogik ausgebildet ist, um einen Betriebszustand von einem von den redundanten Elementen (204, 206) zu detektieren und den Betriebszustand an die Steuereinheit (208) zu kommunizieren.

15. Prozesssteuerungssystem (200) nach Anspruch 1, wobei die Steuereinheit (208) einen Detektor beinhaltet, der die Beendigung von Kommunikati-

onen von einem von dem Paar von redundanten Elementen (**204**, **206**) detektiert, um den Ausfall des einen von dem Paar von redundanten Elementen (**204**, **206**) zu detektieren.

16. Prozesssteuerungssystem (**100**) nach Anspruch 1, wobei die Steuereinheit eine Schleifensteuereinheit (**106**) aufweist, wobei die Schleifensteuereinheit eine Steuerlogik beinhaltet, die ein schleifengespeistes digitales Zweidraht-Zweiwege-Kommunikationsprotokoll implementiert; wobei der Kommunikationsbus einen ersten redundanten Kommunikationsbus (**112**) und einen zweiten redundanten Kommunikationsbus (**113**) aufweist und wobei das erste redundante Element den ersten redundanten Kommunikationsbus (**112**) aufweist und das zweite redundante Element den zweiten redundanten Kommunikationsbus (**113**) aufweist; wobei der erste redundante Kommunikationsbus (**112**) und der zweite redundante Kommunikationsbus (**113**) jeweils einen Zweiwege-Kommunikationsbus aufweisen; wobei der erste redundante Kommunikationsbus (**112**) und der zweite redundante Kommunikationsbus (**113**) mit der Schleifensteuereinheit (**106**) gekoppelt sind; wobei die Vielzahl von Feldgeräten (**116**) mit dem ersten redundanten Kommunikationsbus (**112**) und dem zweiten redundanten Kommunikationsbus (**113**) gekoppelt sind; wobei die Steuereinheit (**106**) ausgebildet ist, um einen Ausfall zu detektieren, der einem von dem ersten redundanten Kommunikationsbus (**112**) und dem zweiten redundanten Kommunikationsbus (**113**) zugeordnet ist, und um bei Detektieren des Ausfalls des einen von dem ersten redundanten Kommunikationsbus (**112**) und dem zweiten redundanten Kommunikationsbus (**113**) den anderen von dem ersten redundanten Kommunikationsbus (**112**) und dem zweiten redundanten Kommunikationsbus (**113**) in das Prozesssteuerungssystem funktionsmäßig einzukoppeln.

17. Prozesssteuerungssystem (**300**) nach Anspruch 16, wobei die Vielzahl von Feldgeräten ein erstes redundantes Feldgerät (**304**) beinhaltet, das mit dem ersten redundanten Kommunikationsbus (**302**) verbunden ist, wobei das Prozesssteuerungssystem ferner ein zweites redundantes Feldgerät (**306**) aufweist, das mit dem zweiten redundanten Kommunikationsbus (**303**) gekoppelt ist.

18. Prozesssteuerungssystem (**400**) nach Anspruch 16, wobei eines von der Vielzahl von Feldgeräten (**404**) mit dem ersten redundanten Kommunikationsbus (**402**) und mit dem zweiten redundanten Kommunikationsbus (**403**) gekoppelt.

19. Prozesssteuerungssystem (**100**) nach Anspruch 16, wobei die Schleifensteuereinheit (**106**) ein

Fieldbus-Kommunikationsprotokoll implementiert.

20. Prozesssteuerungssystem (**200**) nach Anspruch 1, wobei die Steuereinheit eine Schleifensteuereinheit (**208**) aufweist, wobei die Schleifensteuereinheit (**208**) eine Steuerlogik beinhaltet, die ein schleifengespeistes, digitales Zweidraht-Zweiwege-Kommunikationsprotokoll implementiert; wobei das Prozesssteuerungssystem ferner eine Zweiwege-Kommunikationsschleife aufweist, die mit der Schleifensteuereinheit (**208**) gekoppelt ist; wobei die Schleifensteuereinheit (**208**) ausgebildet ist, um über die Zweiwege-Kommunikationsschleife einen Ausfall zu detektieren, der einem von dem ersten redundanten Element (**204**) und dem zweiten redundanten Element (**206**) zugeordnet ist, und um bei Detektieren des Ausfalls des einen von dem ersten redundanten Element (**204**) und dem zweiten redundanten Element (**206**) das andere von dem ersten redundanten Element (**204**) und dem zweiten redundanten Element (**206**) funktionsmäßig in das Prozesssteuerungssystem (**200**) einzukoppeln.

21. Verfahren zum Konfigurieren eines Prozesssteuerungssystems (**100**, **200**, **300**, **400**, **500**, **600**, **700**), das auf verteilte Weise Prozesssteuerungsfunktionen in einem Prozess ausführt, wobei das Verfahren die folgenden Schritte aufweist: Vorsehen eines Kommunikationsbusses (**202**), der eine Kommunikationsprozessfunktion in dem Prozesssteuerungssystem ausführt; kommunikatives Verbinden einer Vielzahl von Feldgeräten (**204**) über den Kommunikationsbus (**202**) derart, dass jedes der Feldgeräte (**204**) eine andere Prozessfunktion innerhalb des Prozesses ausführt; gekennzeichnet durch Verwenden des Kommunikationsbusses (**202**), um die Operation der Vielzahl von Feldgeräten (**204**) zu überwachen; Verwenden eines Paares von redundanten Elementen, die ein erstes redundantes Element (**204**) und ein zweites redundantes Element (**206**) beinhalten, innerhalb des Prozesses, um die gleiche Prozessfunktion auszuführen; Verwenden des Kommunikationsbusses (**202**), um einen Ausfall von einem von den redundanten Elementen zu detektieren; und funktionsmäßiges Einkoppeln des anderen von den redundanten Elementen in das Prozesssteuerungssystem in Reaktion auf den Ausfall von dem einen von den redundanten Elementen.

22. Verfahren nach Anspruch 21, wobei der Kommunikationsbus einen ersten redundanten Kommunikationsbus (**402**) und einen zweiten redundanten Kommunikationsbus (**403**) aufweist und wobei das erste redundante Element den ersten redundanten Kommunikationsbus (**402**) aufweist und das zweite redundante Element den zweiten redundanten Kommunikationsbus (**403**) aufweist, und wobei das Ver-

fahren ferner den folgenden Schritt aufweist: Verbinden des ersten redundanten Kommunikationsbusses (**402**) und eines zweiten redundanten Kommunikationsbusses (**403**) mit demselben Feldgerät (**404**).

23. Verfahren nach Anspruch 21, wobei das erste redundante Element eines von der Vielzahl von Feldgeräten (**204**) aufweist und das zweite redundante Element ein weiteres Feldgerät (**206**) aufweist, und wobei das Verfahren ferner den folgenden Schritt aufweist: Verbinden des einen von der Vielzahl von Feldgeräten (**204**) und des weiteren Feldgerätes (**206**) mit dem Kommunikationsbus (**202**).

24. Verfahren nach Anspruch 21, wobei das erste redundante Element einen ersten Funktionsblock (**704**) aufweist, der eine bestimmte Prozessfunktion ausführt, und das zweite redundante Element einen zweiten Funktionsblock (**706**) aufweist, der die bestimmte Prozessfunktion ausführt, und wobei das Verfahren ferner den folgenden Schritt aufweist: alternatives kommunikatives Koppeln entweder des ersten (**704**) oder des zweiten Funktionsblocks (**706**) innerhalb einer Prozesssteuerungsschleife des Prozesses.

25. Verfahren nach Anspruch 24, das ferner den folgenden Schritt aufweist: Anordnen des ersten (**704**) und des zweiten Funktionsblocks (**706**) in verschiedenen von der Vielzahl von Feldgeräten.

26. Verfahren nach Anspruch 24, das ferner den folgenden Schritt aufweist:  
kommunikatives Verbinden eines Steereinheit-Funktionsblocks (**708**) mit dem ersten (**704**) und dem zweiten Funktionsblock (**706**), um den Ausfall von einem von dem ersten (**704**) und dem zweiten Funktionsblock (**706**) zu detektieren.

27. Verfahren nach Anspruch 21, wobei das erste redundante Element eine erste Schleife unter Einschluss eines ersten redundanten Kommunikationsbusses (**302**) in Verbindung mit einem von den Feldgeräten (**304**) aufweist, und das zweite redundante Element eine redundante Schleife unter Einschluss eines zweiten redundanten Kommunikationsbusses (**303**) in Verbindung mit einem redundanten Feldgerät (**306**) aufweist, und wobei das Verfahren ferner den folgenden Schritt aufweist:  
funktionsmäßiges Einkoppeln von nur einer von der ersten Schleife oder der redundanten Schleife innerhalb des Prozesssteuerungssystems (**300**) zu einem bestimmten Zeitpunkt.

Es folgen 10 Blatt Zeichnungen

Anhängende Zeichnungen

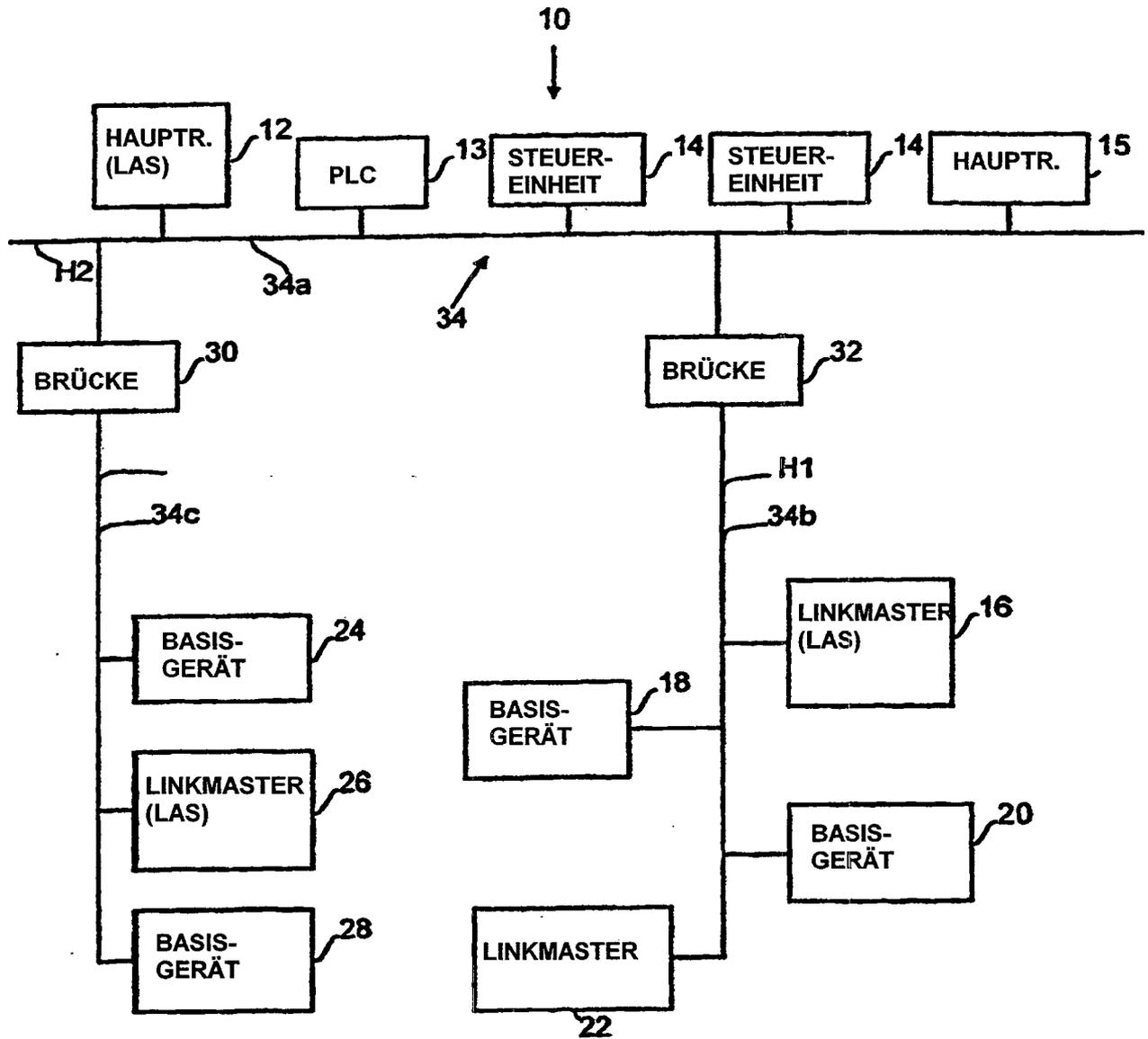


FIG. 1

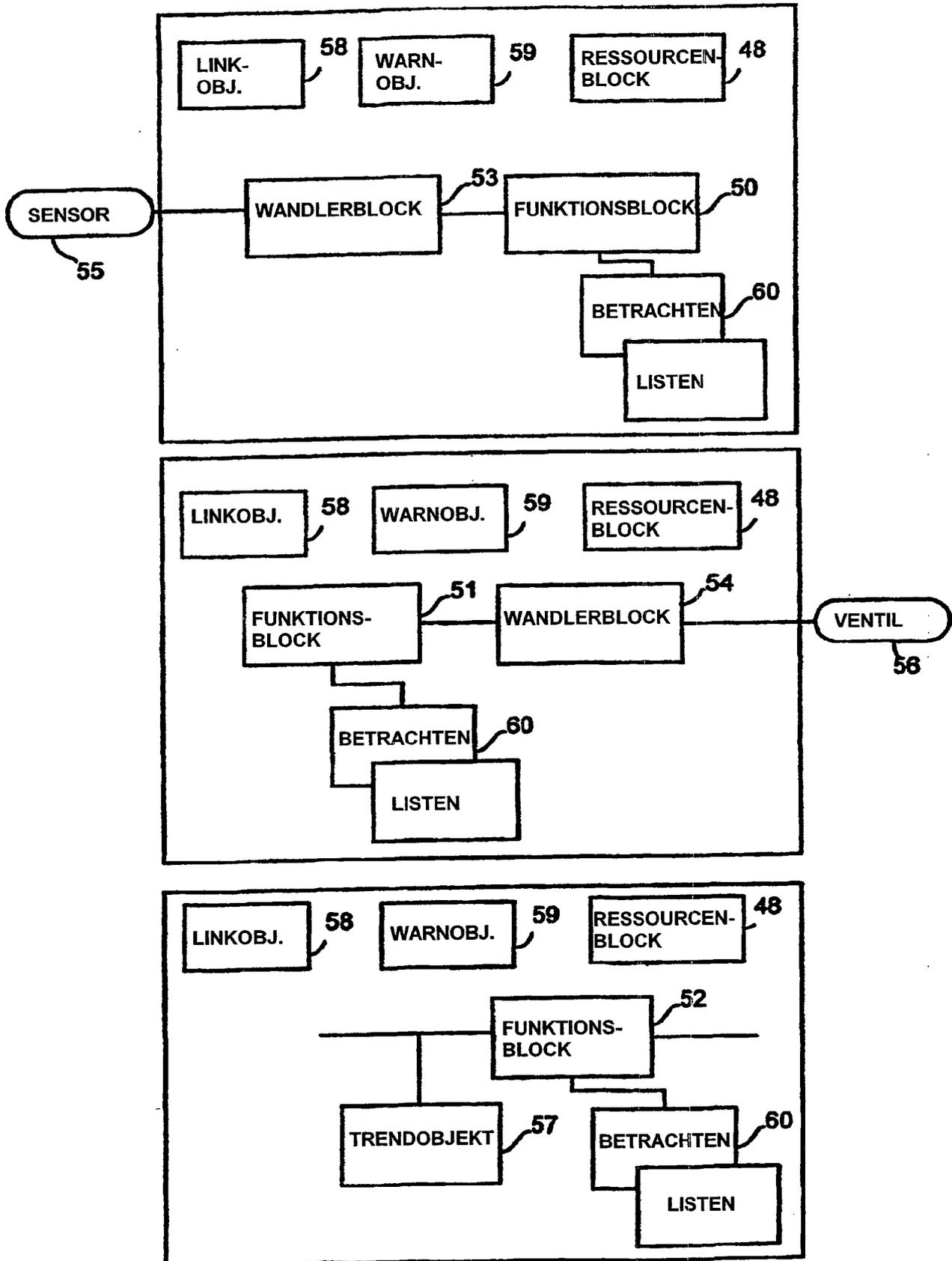


FIG. 2

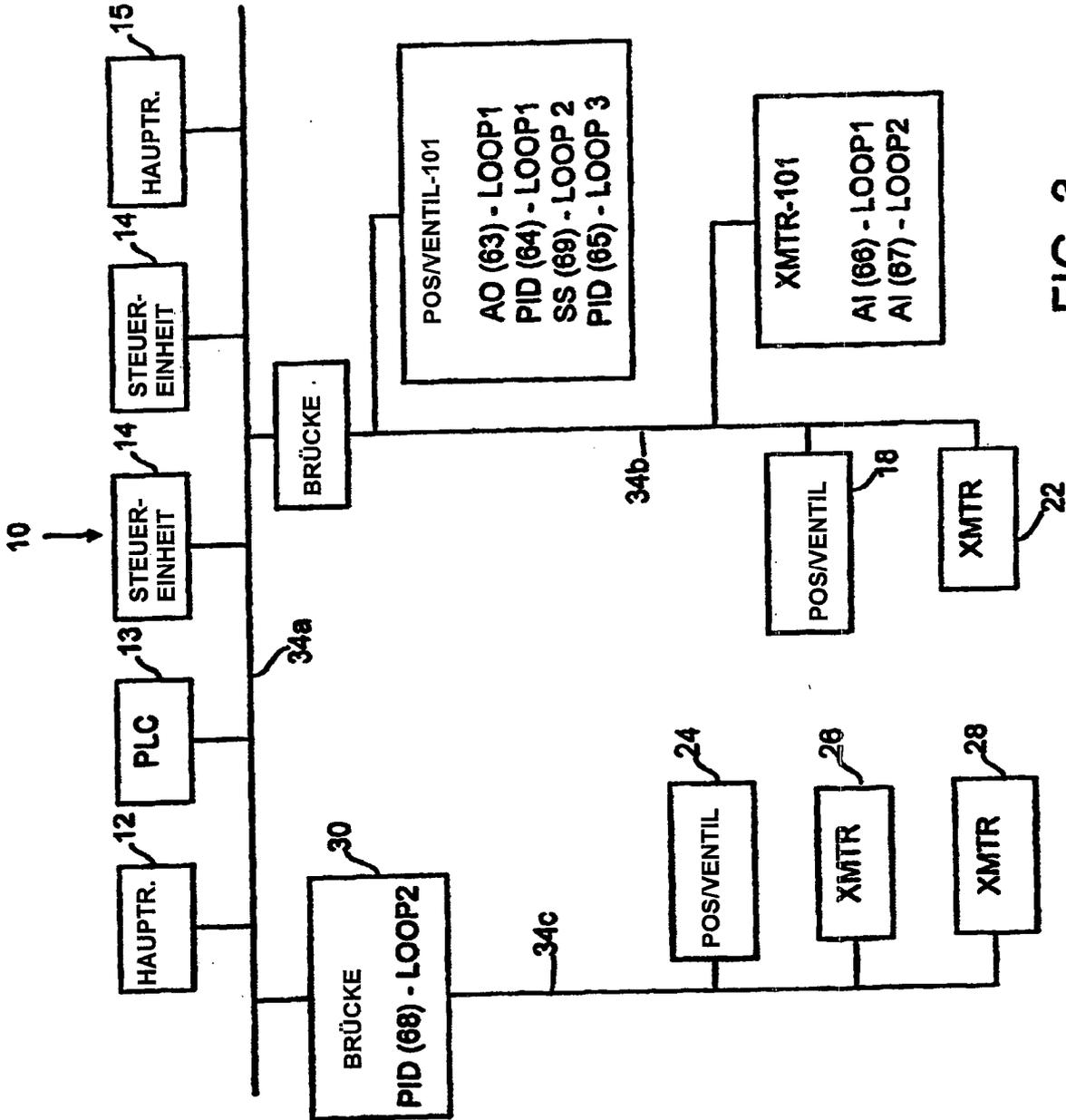


FIG. 3

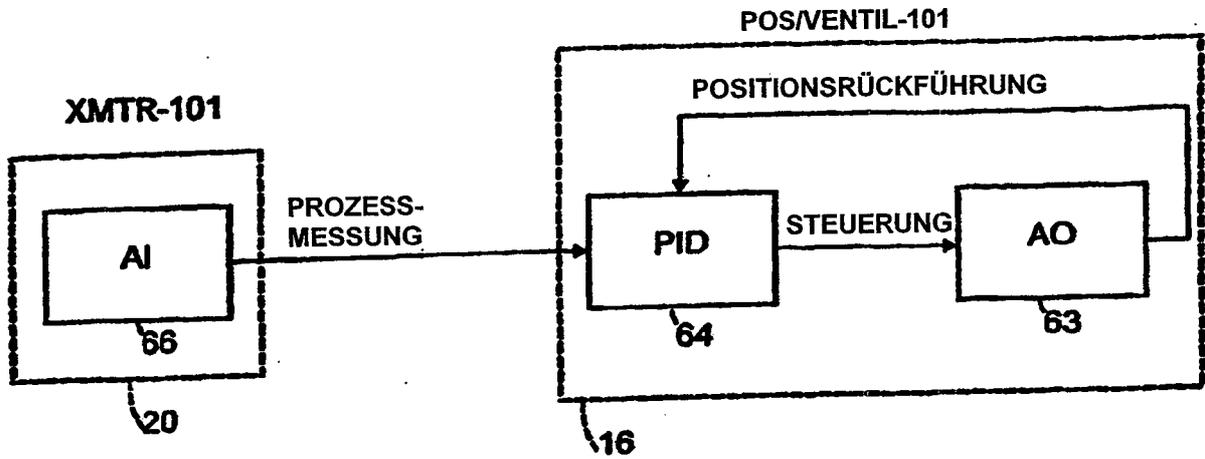


FIG. 4

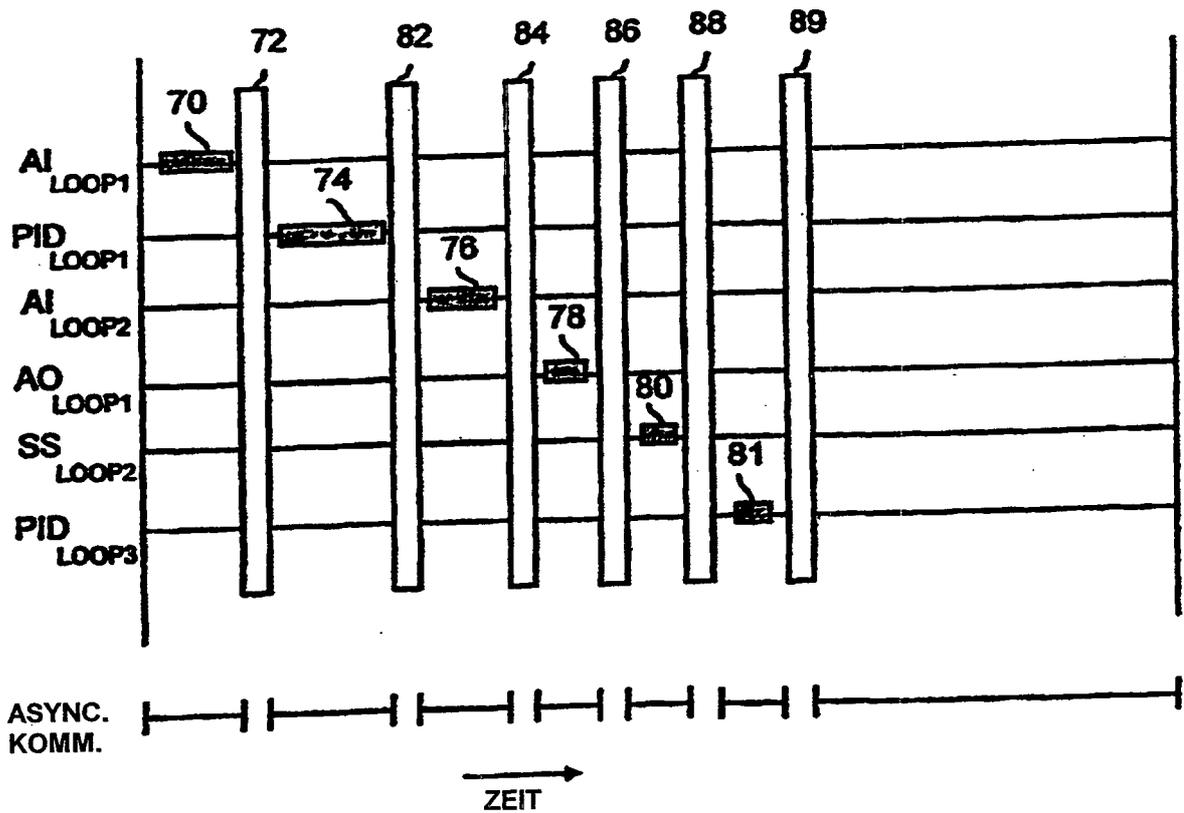


FIG. 5

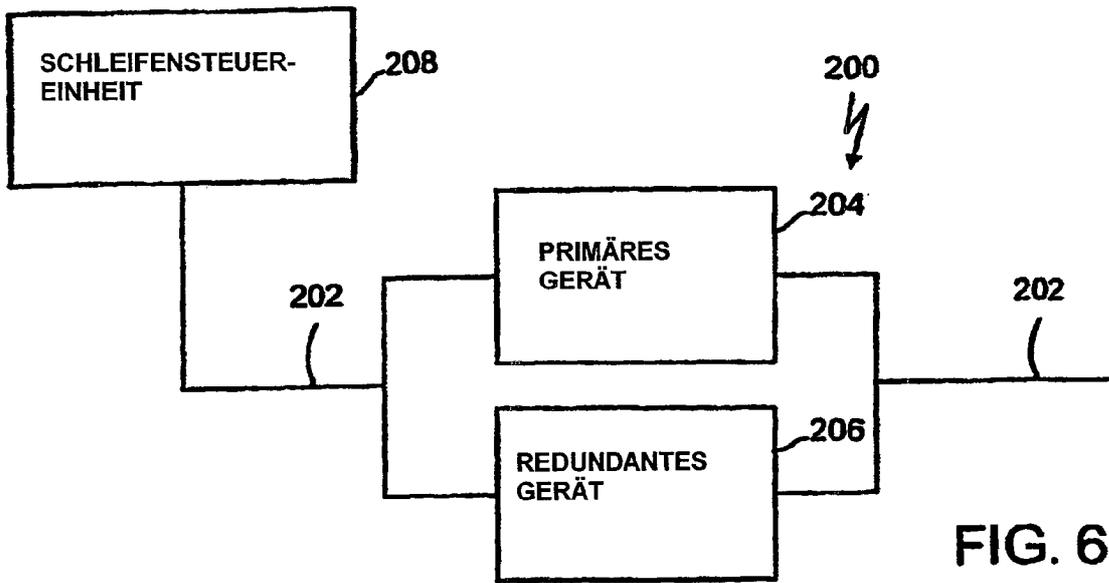


FIG. 6

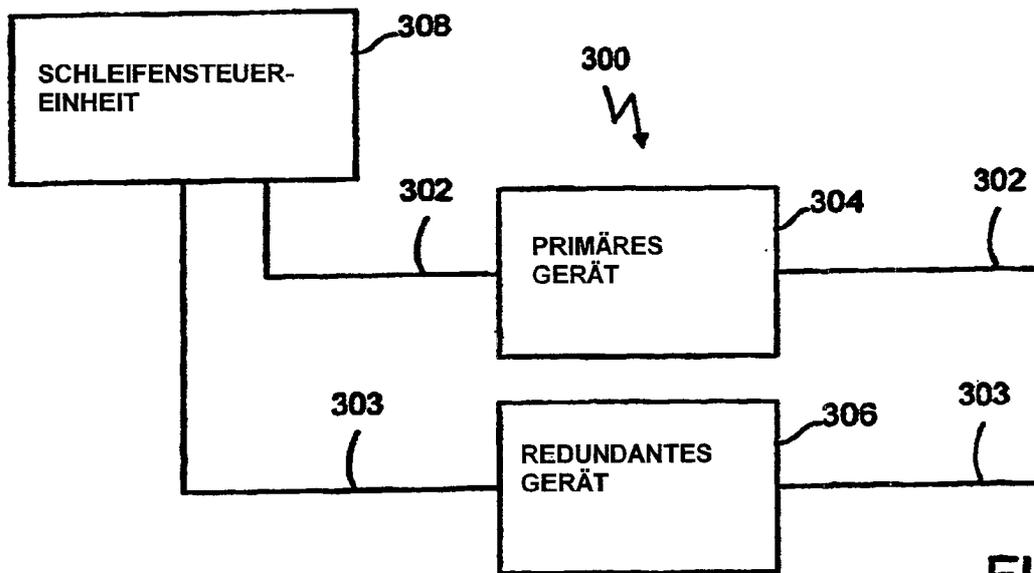


FIG. 7

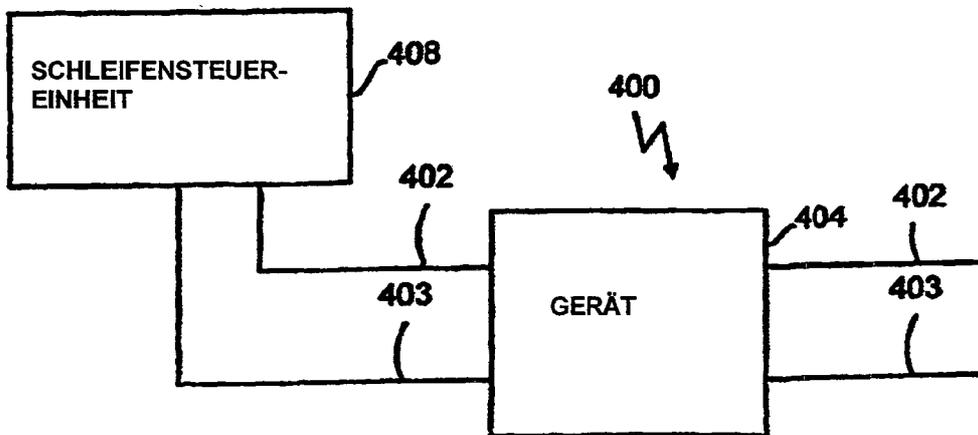


FIG. 8

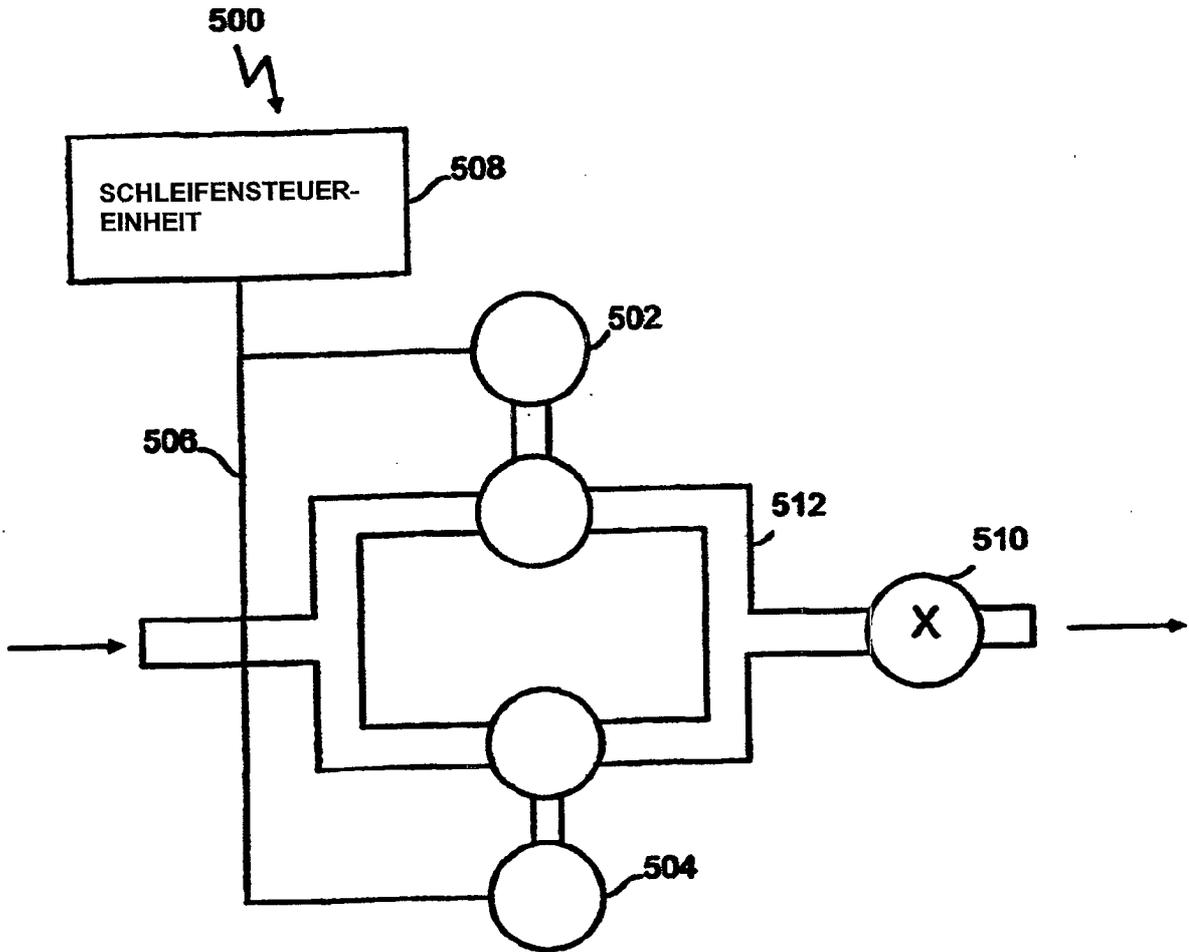


FIG. 9

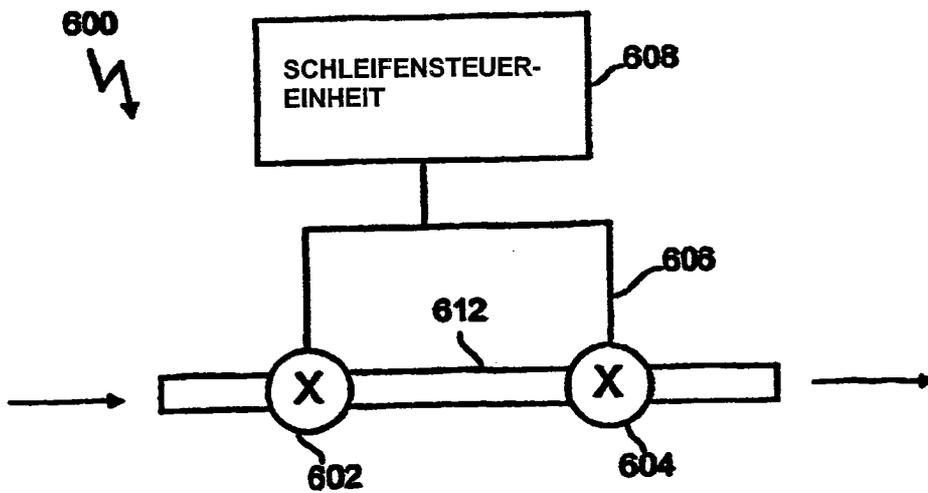


FIG. 10

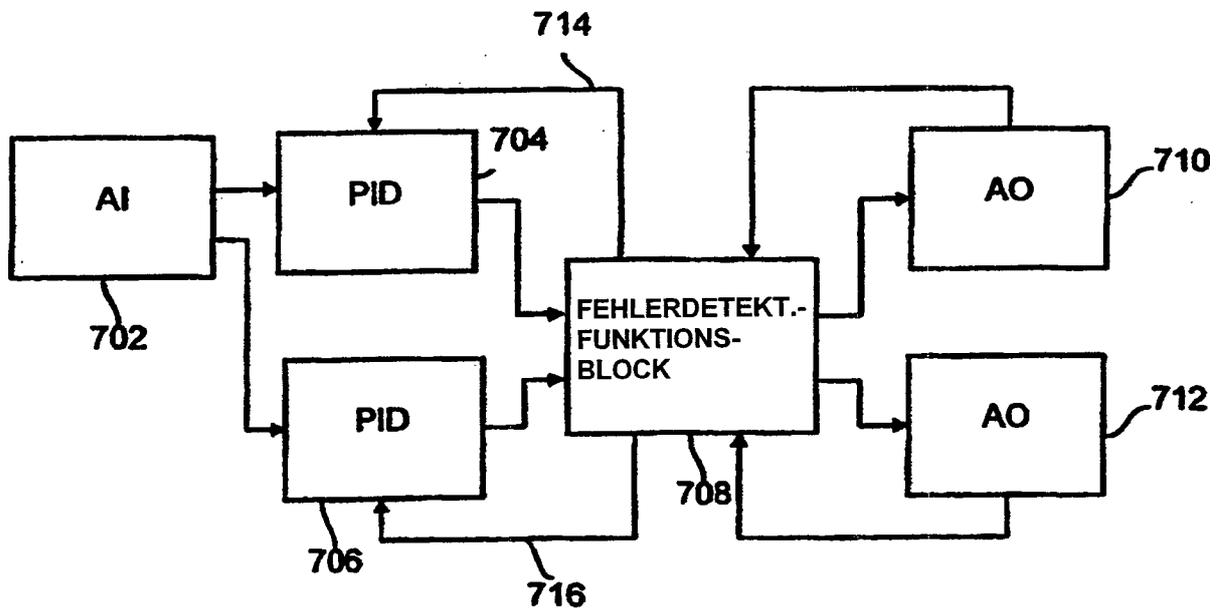


FIG. 11

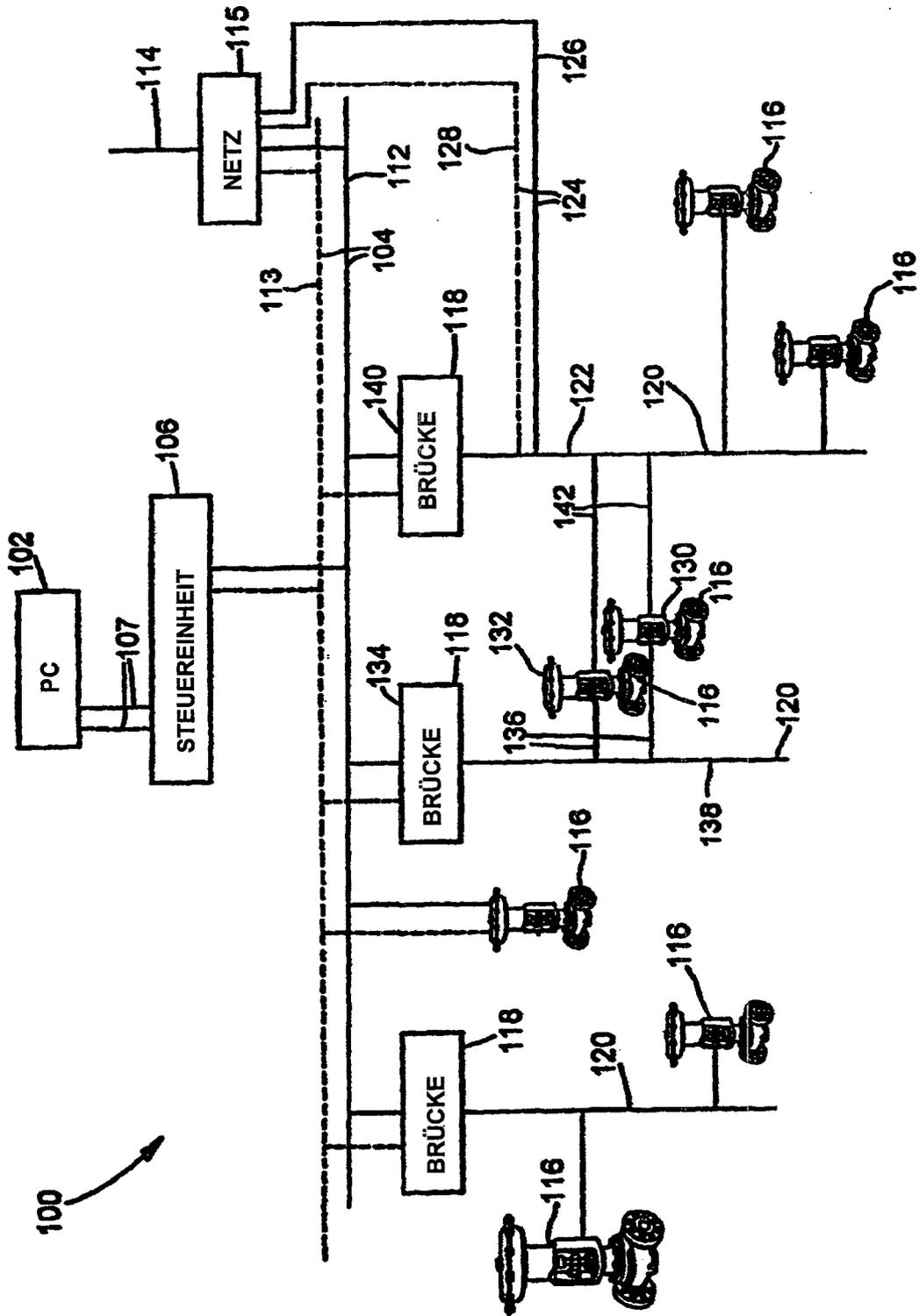


FIG. 12

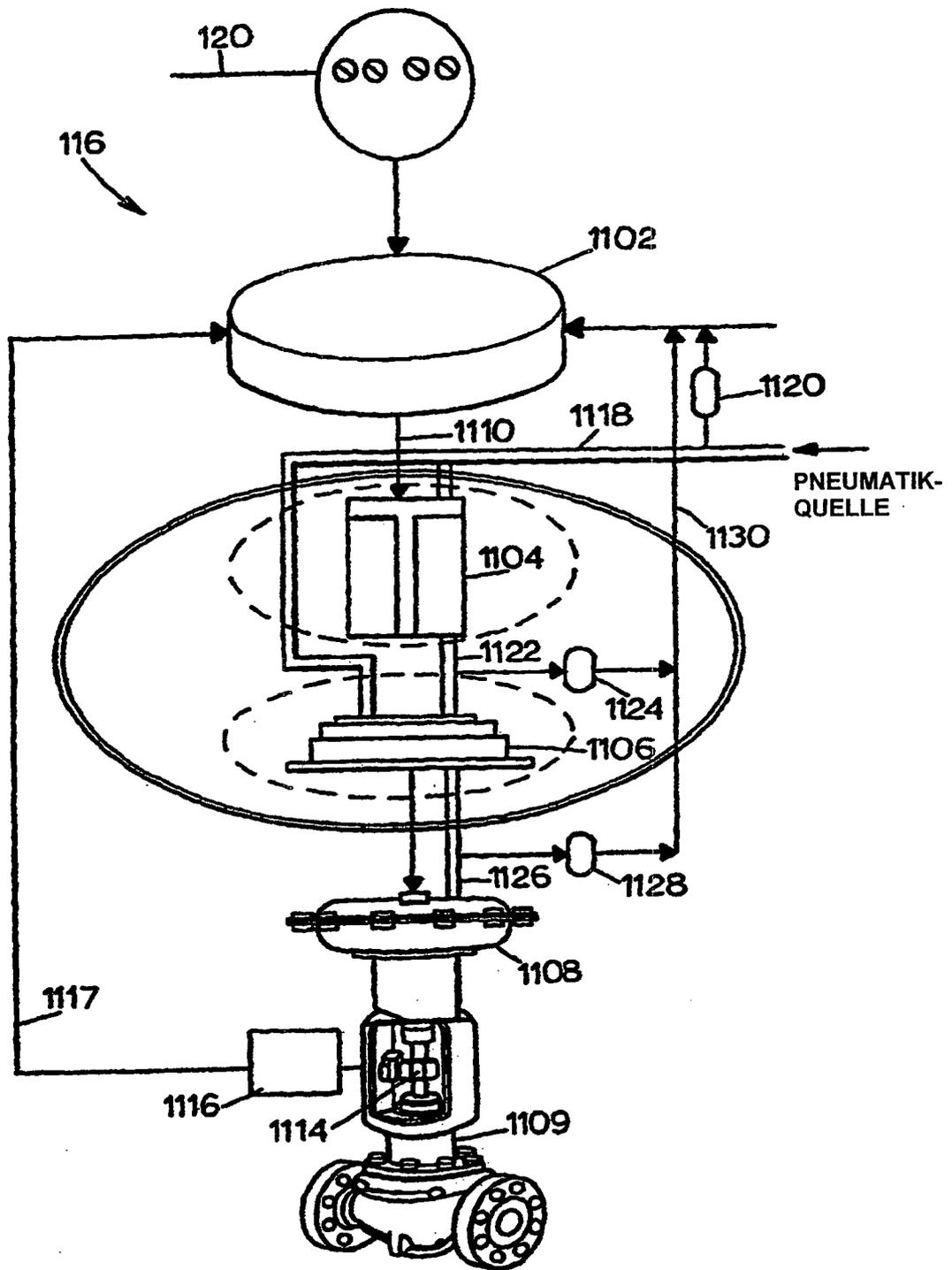


FIG. 13

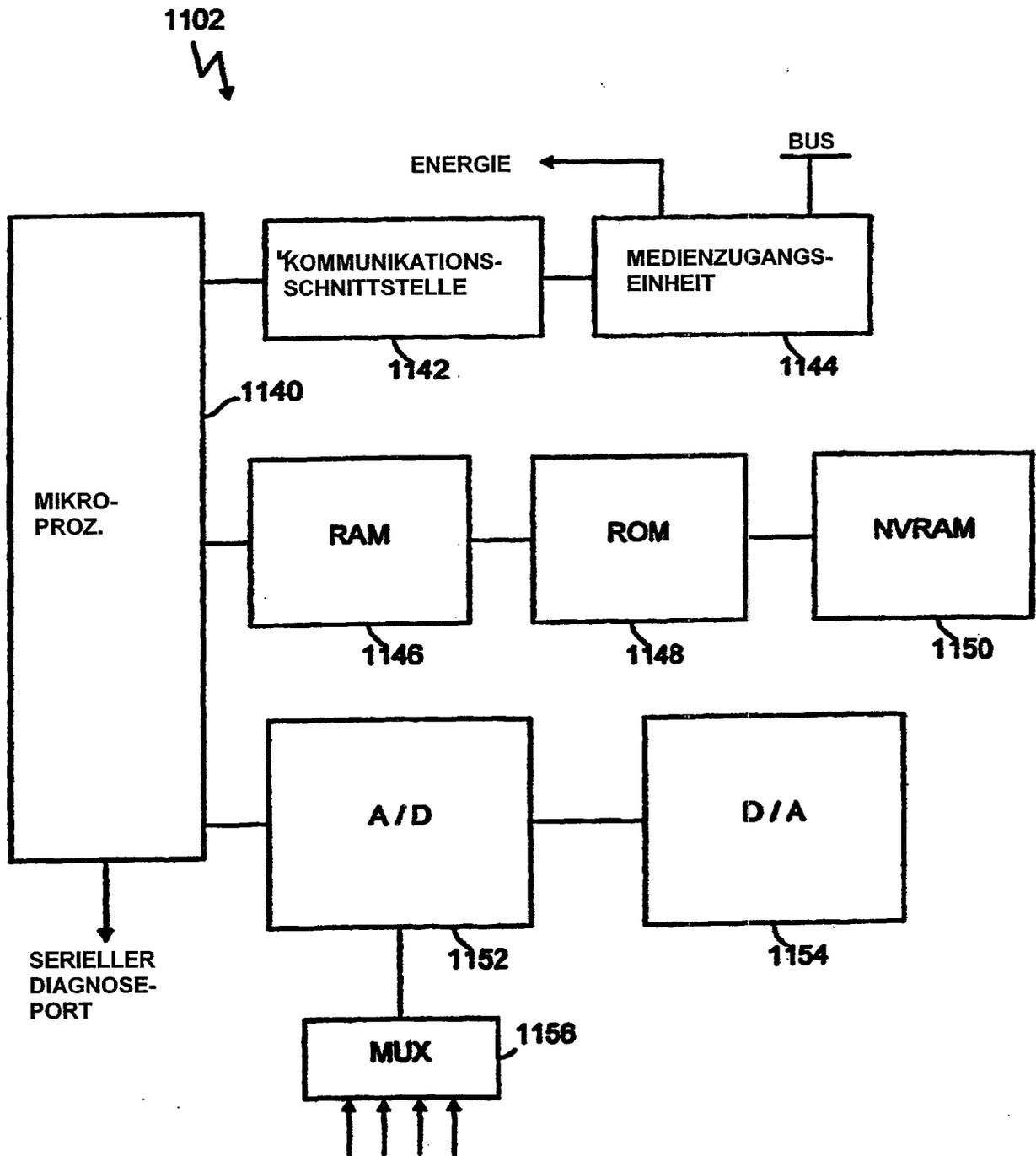


FIG. 14