



(19)  
**Bundesrepublik Deutschland**  
**Deutsches Patent- und Markenamt**

(10) **DE 101 02 205 B4 2007.04.19**

(12)

## Patentschrift

(21) Aktenzeichen: **101 02 205.0**  
 (22) Anmeldetag: **18.01.2001**  
 (43) Offenlegungstag: **26.07.2001**  
 (45) Veröffentlichungstag  
 der Patenterteilung: **19.04.2007**

(51) Int Cl.<sup>8</sup>: **G06F 15/177 (2006.01)**

Innerhalb von drei Monaten nach Veröffentlichung der Patenterteilung kann nach § 59 Patentgesetz gegen das Patent Einspruch erhoben werden. Der Einspruch ist schriftlich zu erklären und zu begründen. Innerhalb der Einspruchsfrist ist eine Einspruchsgebühr in Höhe von 200 Euro zu entrichten (§ 6 Patentkostengesetz in Verbindung mit der Anlage zu § 2 Abs. 2 Patentkostengesetz).

(30) Unionspriorität:  
**09/488335      20.01.2000      US**

(72) Erfinder:  
**Dove, Andrew Philip, Austin, Texas, US; Gardner, David Richard, Austin, Texas, US**

(73) Patentinhaber:  
**Fisher-Rosemount Systems, Inc., Austin, Tex., US**

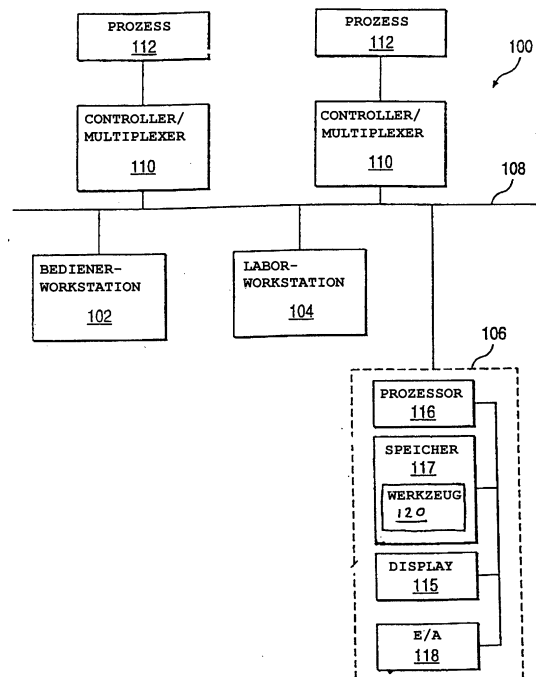
(56) Für die Beurteilung der Patentfähigkeit in Betracht  
 gezogene Druckschriften:  
**US 59 95 916 A**  
**EP 07 51 648 A2**  
**EP 07 18 727 A2**

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

(54) Bezeichnung: **Verfahren und Vorrichtung zum Konfigurieren und Verwalten eines Prozeßsteuerungsnetzes**

(57) Hauptanspruch: Verfahren zum Konfigurieren und Verwalten eines Prozeßsteuerungsnetzes, wobei das Prozeßsteuerungsnetz einen Computer mit einem Prozessor und einem Speicher aufweist, wobei das Verfahren die folgenden Schritte aufweist:

Versorgen des Computers mit auf ein räumliches Layout einer Anlage bezogenen Informationen;  
 Versorgen des Computers mit auf die Konfiguration des Prozeßsteuerungsnetzes bezogenen Informationen;  
 Präsentieren eines auf das räumliche Layout der Anlage bezogenen Layouts der Konfiguration des Prozeßsteuerungsnetzes,  
 gekennzeichnet durch  
 ein Analysieren des Layouts des Prozeßsteuerungsnetzes unter Bezugnahme auf das physische Layout der Anlage, um sicherzustellen, daß das Layout des Prozeßsteuerungsnetzes mit den Kriterien eines Standardprotokolls für Prozeßsteuerungsnetzwerke übereinstimmt, welches nicht auf der Basis von TCP/IP beruht.



## Beschreibung

**[0001]** Die Erfindung betrifft allgemein Prozeßsteuerungsnetze und speziell das Konfigurieren und Verwalten von Prozeßsteuerungsnetzen.

### Stand der Technik

**[0002]** Die EP 0 751 648 A2 zeigt ein Navigations- und Betrachtungssystem für die Netzwerk-Administration, welches das dreidimensionale physische Layout von Computern in einem (Büro-) Gebäude darstellen kann. Zusätzlich gibt es dem Administrator die Möglichkeit, die Domainzugehörigkeit und somit die Topologie des Netzwerkes sowie eine grafische Repräsentation bestimmter aktiver Verbindungen zwischen den Computern zu betrachten.

**[0003]** Die EP 0 718 727 A2 zeigt ein Verfahren zur Programmierung eines verteilten Prozeßsteuerungsnetzwerkes, welches eine grafische Benutzeroberfläche aufweist. Damit lassen sich die diversen verteilten Komponenten des Prozeßsteuerungsnetzwerkes zentral konfigurieren bzw. steuern. Dabei ist aber das Prozeßsteuerungsnetzwerk allein durch seine logische Struktur, also seine Topologie, innerhalb der Benutzeroberfläche repräsentiert.

**[0004]** Aus der US 5 995 916 A ist die Überwachung und die Anzeige von diagnostischer Information in einem Prozeßsteuerungssystem bekannt. Das Prozeßsteuerungssystem berücksichtigt diagnostische Informationen, die sich auf alle Einrichtungen beziehen und präsentiert diese Informationen einem Systemanwender in einer Art, dass eine Betriebssteuerungsstrategie und die diagnostische Information so präsentiert werden, als ob alle Steuerungsaktionen und die diagnostische Information an einem einzelnen Ort ausgeführt bzw. generiert werden.

**[0005]** Große Prozesse wie etwa chemische, Erdöl- und andere Herstellungs- und Raffinationsprozesse weisen zahlreiche Feldeinrichtungen auf, die an verschiedenen Stellen innerhalb einer Anlage angeordnet sind, um Prozeßparameter zu messen und zu steuern, die dadurch die Steuerung des Prozesses bewirken. Diese Einrichtungen können beispielsweise Sensoren wie etwa Temperatur-, Druck- und Durchflußmengensensoren sowie Stellelemente wie etwa Absperrorgane bzw. Ventile und Schalter sein. Historisch verwendete die Prozeßsteuerungsindustrie manuelle Vorgänge wie etwa das manuelle Ablesen von Werten und Manometern, das Drehen von Ventilrädern usw., um die Messung durchzuführen und Feldeinrichtungen innerhalb eines Prozesses zu steuern.

**[0006]** Heute wird die Steuerung des Prozesses häufig unter Anwendung von Mikroprozessorbasierten Controllern, Computern oder Workstations imple-

mentiert, die den Prozeß überwachen, indem sie Befehle und Daten an Hardwareeinrichtungen senden bzw. von diesen empfangen, um entweder einen bestimmten Aspekt des Prozesses oder den Prozeß insgesamt zu steuern. Die speziellen Prozeßsteuerungsfunktionen, die durch Softwareprogramme in diesen Mikroprozessoren, Computern oder Workstations implementiert werden, können durch Programmieren individuell entworfen, modifiziert oder geändert werden, während gleichzeitig keine Modifikationen der Hardware erforderlich sind. Beispielsweise kann ein Techniker ein Programm schreiben lassen, damit der Controller einen Fluidpegel von einem Pegelsensor in einem Tank abliest, den Tankpegel mit einem vorbestimmten Sollpegel vergleicht und dann, basierend darauf, ob der gemessene Pegel niedriger oder höher als der vorbestimmte Sollpegel war, ein Füllventil öffnet oder schließt. Die Parameter können leicht geändert werden, indem eine ausgewählte Ansicht des Prozesses angezeigt und dann das Programm unter Nutzung der ausgewählten Ansicht modifiziert wird. Typischerweise ändert der Ingenieur Parameter durch Anzeigen und Modifizieren einer technischen Ansicht des Prozesses.

**[0007]** Der Controller, Computer oder die Workstation speichert und implementiert ein zentralisiertes und häufig komplexes Steuerungsschema, um Messungen und die Steuerung von Prozeßparametern nach einem Gesamtsteuerungsschema zu bewirken. Gewöhnlich ist aber das implementierte Steuerungsschema Eigentum des Herstellers der Feldeinrichtung, so daß es schwierig und teuer ist, das Prozeßsteuerungssystem zu erweitern, auszubauen, umzuprogrammieren und/oder zu warten, weil der Hersteller der Feldeinrichtung auf eine integrale Weise involviert sein muß, um jede dieser Aktivitäten auszuführen. Außerdem können die Geräte, die verwendet oder miteinander verbunden werden können, aufgrund der Eigentumsverhältnisse der Feldeinrichtung und dadurch, daß der Hersteller bestimmte Einrichtungen oder Funktionen von Einrichtungen, die von anderen Herstellern stammen, eventuell nicht unterstützt, eingeschränkt sein.

**[0008]** Um einige der Probleme zu überwinden, die mit der Verwendung von herstellereinspezifischen Feldeinrichtungen einhergehen, hat die Prozeßsteuerungs-Industrie eine Reihe von offenen Standard-Kommunikationsprotokollen entwickelt, die beispielsweise die HART<sup>®</sup>-, DE-, PROFIBUS<sup>®</sup>-, WORLDFIP<sup>®</sup>-, LONWORKS<sup>®</sup>-, Device-Net<sup>®</sup> und CAN-Protokolle umfassen. Diese Standardprotokolle ermöglichen es, von verschiedenen Herstellern stammende Feldeinrichtungen gemeinsam innerhalb derselben Prozeßsteuerungsumgebung zu verwenden. Theoretisch kann jede Feldeinrichtung, die mit einem dieser Protokolle übereinstimmt, innerhalb eines Prozesses verwendet werden, um mit einem Prozeßsteuerungssystem oder einem anderen Control-

ler, der das Protokoll unterstützt, zu kommunizieren oder davon gesteuert zu werden, und zwar auch dann, wenn die Feldeinrichtungen von verschiedenen Herstellern stammen.

**[0009]** Zur Implementierung von Steuerfunktionen weist jede Prozeßsteuerungseinrichtung einen Mikroprozessor auf, der imstande ist, eine oder mehrere Grundsteuerfunktionen auszuführen, sowie die Fähigkeit hat, mit anderen Prozeßsteuerungseinrichtungen unter Nutzung eines offenen Standard-Protokolls zu kommunizieren. Auf diese Weise können von verschiedenen Herstellern stammende Feldeinrichtungen innerhalb einer Prozeßsteuerungsschleife miteinander verbunden werden, um miteinander zu kommunizieren und eine oder mehrere Prozeßsteuerfunktionen oder Steuerschleifen auszuführen. Ein weiteres Beispiel eines offenen Kommunikationsprotokolls, das es erlaubt, daß von verschiedenen Herstellern stammende Einrichtungen über einen Standardbus zusammenwirken und miteinander kommunizieren, um innerhalb eines Prozesses eine dezentralisierte Steuerung durchzuführen, ist das FOUNDATION Fieldbus-Protokoll (nachstehend "Fieldbus-Protokoll") der Fieldbus Foundation. Das Fieldbus-Protokoll ist ein vollständig digitales, Zweidrahtschleifenprotokoll.

**[0010]** Bei Anwendung dieser Protokolle bezieht sich eine Herausforderung, die mit dem Entwerfen des Prozeßsteuerungssystems oder -netzes einhergeht, auf das tatsächliche physische Layout und die Verbindungen zwischen den verschiedenen Prozeßsteuerungseinrichtungen. Insbesondere gibt jedes dieser Protokolle Einschränkungen von Werten für die physischen Charakteristiken an, innerhalb denen ein Prozeßsteuerungssystem arbeiten muß, um mit dem Standard übereinzustimmen. Diese Einschränkungen umfassen den Spannungsabfall über Kommunikationsabschnitten, die Länge der Stichstrecke, die Gesamtkabellänge, die Gesamtstromentnahme und die Gesamtzahl von Prozeßsteuerungseinrichtungen an einem bestimmten Netzknoten. Der physische Ort von Behältern, Rohrleitungen, Pumpen, Motoren und Ventilen sowie von Controllern und Bedienerstationen ist ebenfalls mit Einschränkungen verbunden, die zu berücksichtigen sind, wenn das Prozeßsteuerungssystem oder -netz konfiguriert wird. Die Beziehungen zwischen diesen Einschränkungen sind wichtig und auf der Basis der Werte der Einschränkungen variabel. Wenn das Prozeßsteuerungssystem oder -netz konfiguriert und in Gebrauch ist, kann die Verwaltung des Systems aufgrund der Komplexität der meisten Raffinations- und Herstellungsanlagen umständlich sein.

**[0011]** Zusätzlich zu der Ausführung von Steuerungsprozessen gibt es Softwareprogramme, die ebenfalls die Prozesse überwachen und eine Ansicht derselben liefern, Feedback in Form eines Bediener-

displays oder einer Bedieneransicht liefern, die den Status bestimmter Prozesse betrifft. Die überwachenden Softwareprogramme liefern ferner einen Alarm, wenn ein Problem auftritt. Manche Programme zeigen einem Bediener Befehle oder Vorschläge an, wenn ein Problem auftritt. Der Bediener, der für den Steuerungsprozeß verantwortlich ist, muß den Prozeß von seinem Standpunkt aus betrachten und das Problem rasch beseitigen können. Ein Display oder eine Konsole ist typischerweise als die Schnittstelle zwischen dem Mikroprozessor-basierten Controller oder Computer, der die Prozeßsteuerungsfunktion ausführt, und dem Bediener sowie auch zwischen dem Programmierer oder Techniker und dem Mikroprozessor-basierten Controller oder Computer, der die Prozeßsteuerungsfunktion ausführt, vorgesehen.

**[0012]** Systeme, die in Prozeßsteuerungsumgebungen Funktionen ausführen, überwachen, steuern und rückführen, werden typischerweise mittels Software implementiert, die in problemorientierten Computerprogrammiersprachen wie etwa Basic, Fortran oder C geschrieben sind, und auf einem Computer oder Controller ausgeführt. Diese problemorientierten Sprachen sind zwar für die Programmierung der Prozeßsteuerung effektiv, werden aber von Verfahrenstechnikern, Regelungstechnikern, Bedienern und Überwachungspersonen gewöhnlich nicht verstanden. Für solche Personen sind höherstufige graphische Displaysprachen entwickelt worden, beispielsweise kontinuierliche Funktionsblock- und Kettenlogik. Jeder von den Technikern, Wartungsleuten, Bedienern, Laborpersonal und dergleichen benötigt daher eine graphische Ansicht der Elemente des Prozeßsteuerungssystems, die es ihnen ermöglicht, das System auf eine in bezug auf ihre Verantwortlichkeiten relevante Weise zu betrachten.

**[0013]** Die graphischen Ansichten der Elemente des Prozeßsteuerungssystems werden ohne Korrelation mit dem räumlichen Layout der Anlage bereitgestellt und zeigen nur logische Verknüpfungen der Einrichtungen und Funktionen. Beispielsweise kann ein Prozeßsteuerungsprogramm in Fortran geschrieben sein und zwei Eingangswerte benötigen, den Mittelwert der Eingangswerte berechnen und einen Ausgangswert erzeugen, der gleich dem Mittelwert der beiden Eingangswerte ist. Dieses Programm könnte als die MITTELWERT-Funktion bezeichnet werden und kann aufgerufen und von den Verfahrenstechnikern über eine graphische Anzeige angesehen werden. Eine typische graphische Anzeige kann aus einer Box bestehen, die zwei Eingänge, einen Ausgang und ein Etikett, das die Box als MITTELWERT bezeichnet, bestehen. Ein davon verschiedenes Programm kann verwendet werden, um die graphische Darstellung derselben Funktion für einen Bediener zu schaffen, so daß dieser den Mittelwert betrachten kann. Bevor das System an den Kunden ausgeliefert

wird, werden diese Softwareprogramme in eine Bibliothek von vordefinierten, vom Anwender wählbaren Merkmalen gestellt. Die Programme sind durch Funktionsblöcke identifiziert. Ein Anwender kann dann eine Funktion aufrufen und die vordefinierten graphischen Darstellungen auswählen, die durch Boxen veranschaulicht sind, um verschiedene Ansichten für den Bediener, den Techniker usw. zu erzeugen, indem einer von einer Vielzahl von Funktionsblöcken aus der Bibliothek ausgewählt wird zur Verwendung bei der logischen Definition einer Prozeßsteuerungslösung, anstatt daß ein vollständig neues Programm beispielsweise in Fortran zu entwickeln ist.

**[0014]** Eine Gruppe von standardisierten Funktionen, die jeweils durch einen zugehörigen Funktionsblock bezeichnet sind, kann in einer Steuerungsbibliothek gespeichert sein. Ein Designer, der mit einer solchen Bibliothek ausgestattet ist, kann Prozeßsteuerungslösungen entwerfen, indem er auf einem Computermonitor verschiedene Funktionen oder Elemente logisch miteinander verknüpft, die mit den durch Boxen repräsentierten Funktionsblöcken ausgewählt werden, um bestimmte Aufgaben auszuführen. Der Mikroprozessor oder Computer ordnet jede der durch die Funktionsblöcke definierten Funktionen bzw. jedem solchen Element vordefinierte Schablonen zu, die in der Bibliothek gespeichert sind, und setzt jede der Programmfunktionen oder jedes der Programmelemente entsprechend den von dem Designer gewünschten Verknüpfungen in Beziehung zueinander. Ein Designer entwirft ein vollständiges Prozeßsteuerungsprogramm unter Verwendung von logischen Ansichten vordefinierter Funktionen, ohne jemals den Entwurf mit den räumlichen Dimensionen der Raffinations- oder Fertigungsanlage zu korrelieren.

**[0015]** Eine Herausforderung, die mit den bereitgestellten graphischen Darstellungen verbunden ist, ist die, daß nur logische Verknüpfungen gezeigt werden. Derzeit wird das physische Layout der Anlage nicht mit der Konfiguration des Prozeßsteuerungssystems korreliert, und bei der Verwaltung des Systems kann nicht darauf zurückgegriffen werden. Bei der Konfiguration des Prozeßsteuerungssystems müssen räumliche Informationen manuell gemessen und in das Werkzeug eingegeben werden. Bei der Verwaltung des Prozeßsteuerungssystems muß der physische Ort von Einrichtungen und Controllern manuell bestimmt werden, wodurch häufig mehr Zeit erforderlich ist, um ein Problem zu korrigieren oder das Prozeßsteuerungssystem zu verwalten.

**[0016]** Was notwendig ist, ist ein Verfahren zum Konfigurieren eines Prozeßsteuerungssystems, das sowohl das physische Layout der Anlage berücksichtigt als auch Bedienern des Systems den raschen Zugriff auf die räumliche Lage von Prozeßsteuerungseinrichtungen und Controllern erlaubt.

## Aufgabenstellung

**[0017]** Aufgabe der vorliegenden Erfindung ist es also, ein System zur Konfiguration und Überwachung eines Netzwerkes zu schaffen, das die Funktionsfähigkeit des Netzwerkes für die Prozeßsteuerung in Anbetracht der unterschiedlichen Kompatibilitätsanforderungen seiner Komponenten sicherstellt. Diese Aufgabe wird durch die Gegenstände der Ansprüche 1, 5 und 9 gelöst.

**[0018]** Die Erfindung richtet sich auf die Nutzung von räumlichen Informationen einer Anlage, um ein Prozeßsteuerungssystem zu konfigurieren und zu verwalten, das in der Anlage vorhanden ist. Das Prozeßsteuerungssystem stimmt mit einem Standardprotokoll für Prozeßsteuerungsnetzwerke überein, welches nicht auf der Basis von TCP/IP beruht. Ein solches System erlaubt auf vorteilhafte Weise das effiziente Entwerfen und die Nutzung eines Prozeßsteuerungssystems, während gleichzeitig sichergestellt wird, daß die physischen Charakteristiken des Systems dem Standard entsprechen. Außerdem ermöglicht ein solches System auch auf vorteilhafte Weise eine effizientere Diagnose, Online-Fehlersuche, Alarmverwaltung und Wartung von Einrichtungen.

**[0019]** Das Werkzeug kann fakultativ eine automatische Erzeugung des Layouts des Prozeßsteuerungsnetzes ermöglichen, das auf das räumliche Layout der Anlage angewandt wird.

**[0020]** Bei einer anderen Ausführungsform wird das Werkzeug benutzt, um das Layout des Prozeßsteuerungsnetzes, das auf das physische Layout der Anlage angewandt wird, zu analysieren und sicherzustellen, daß das Layout des Netzes mit den Kriterien eines Standardprotokolls wie etwa des Fieldbus-Protokolls übereinstimmt.

**[0021]** Das Werkzeug kann fakultativ blinkende Darstellungen von Einrichtungen bereitstellen, um aktive Alarmlinien in dem Netz zu bezeichnen.

**[0022]** Bei einer anderen Ausführungsform ist das Prozeßsteuerungsnetz so konfiguriert, daß es zuerst logische Verknüpfungen benutzt, und zu einem späteren Zeitpunkt wird die Konfiguration dann auf das räumliche Layout der Anlage angewandt und zur Verwaltung des Prozeßsteuerungsnetzes unter Nutzung der auf das Netzlayout angewandten räumlichen Informationen angewandt.

## Ausführungsbeispiel

**[0023]** Die Erfindung wird nachstehend auch hinsichtlich weiterer Merkmale und Vorteile anhand der Beschreibung von Ausführungsbeispielen unter Bezugnahme auf die beiliegenden Zeichnungen näher

erläutert. Die Zeichnungen zeigen in:

[0024] [Fig. 1a](#) ein schematisches Blockbild eines Prozeßsteuerungssystems, das eine Workstation mit einem Programmierwerkzeug gemäß einer verallgemeinerten Ausführungsform der Erfindung aufweist, die eine räumliche Konfigurations- und Verwaltungsfähigkeit bietet;

[0025] [Fig. 1b](#) ein Blockbild des Controller-/Multiplexer- und Prozeßbereichs des in [Fig. 1a](#) gezeigten Prozeßsteuerungssystems;

[0026] [Fig. 2](#) ein schematisches Blockbild, das die Prozeßsteuerungsumgebung in einer Konfigurationsimplementierung und einer Verwaltungs- oder Ausführungszeitimplementierung zeigt;

[0027] [Fig. 3](#) ein Ablaufdiagramm, das die Operation eines Programmierwerkzeugs gemäß der Erfindung zeigt;

[0028] [Fig. 4](#) eine Bildschirmdarstellung des Konfigurationsbereichs des Programmierwerkzeugs, wobei logische Verknüpfungen zwischen Funktionen und Einrichtungen eines Prozeßsteuerungssystems gezeigt sind; und

[0029] [Fig. 5a](#) bis [Fig. 5c](#) Ansichten der Bildschirmdarstellung des räumlichen Bereichs, wobei physische Verbindungen zwischen Funktionen und Einrichtungen eines Prozeßsteuerungssystems und ihre relativen Positionen in dem räumlichen Layout der Anlage gezeigt sind.

[0030] [Fig. 1a](#) zeigt eine Prozeßsteuerungsumgebung **100** zur Veranschaulichung einer Steuerungsumgebung für die Implementierung eines digitalen Steuerungssystems, eines Prozeßcontrollers oder dergleichen. Die Prozeßsteuerungsumgebung **100** umfaßt eine Bedienerworkstation **102**, eine Laborworkstation **104** und eine Programmierworkstation **106**, die durch ein lokales Datennetz bzw. LAN **108** oder eine andere bekannte Nachrichtenverbindung elektrisch miteinander verbunden sind, um Daten und Steuersignale zwischen den verschiedenen Workstations und einer Vielzahl von Controllern/Multiplexern **110** zu übertragen und zu empfangen. Die Workstations **102**, **104** und **106** sind beispielsweise Computer, die mit der IBM-kompatiblen Architektur übereinstimmen. Die Workstations **102**, **104** und **106** sind durch das LAN **108** mit einer Vielzahl der Controller/Multiplexer **110** verbunden gezeigt, die als elektrische Schnittstellen zwischen den Workstations und einer Vielzahl von Prozessen **112** wirken. Bei einer Vielzahl von verschiedenen Ausführungsformen weist das LAN **108** eine einzelne Workstation auf, die direkt mit einem Controller/Multiplexer **110** verbunden ist, oder weist alternativ eine Vielzahl von Workstations, beispielsweise drei Workstations **102**, **104**

und **106** sowie viele Controller/Multiplexer **110** in Abhängigkeit von den Zielen und Erfordernissen der Prozeßsteuerungsumgebung **100** auf. Bei einigen Ausführungsformen steuert ein einziger Prozeßcontroller/Multiplexer **110** mehrere verschiedene Prozesse **112** oder steuert alternativ einen Teil eines einzelnen Prozesses.

[0031] In der Prozeßsteuerungsumgebung **100** wird eine Prozeßsteuerungsstrategie entwickelt durch Festlegen einer Softwaresteuerungslösung beispielsweise an der Programmierworkstation **106** und Übertragen der Lösung über das LAN **108** zu der Bedienerworkstation **102**, der Laborworkstation **104** und dem Controller/Multiplexer **110** zur Ausführung. Die Bedienerworkstation **102** liefert Schnittstellenanzeigen an die Steuerungs-/Überwachungsstrategie, die in dem Controller/Multiplexer implementiert ist, und kommuniziert mit einem oder mehreren der Controller/Multiplexer **110**, um die Prozesse **112** zu betrachten und Steuerungsattributwerte nach Maßgabe der Erfordernisse der entworfenen Lösung zu ändern. Die Prozesse **112** sind aus einer oder mehreren Feldeinrichtungen gebildet, die intelligente oder herkömmliche (nichtintelligente) Feldeinrichtungen sein können.

[0032] Außerdem überträgt die Bedienerworkstation **102** visuelles und Audiofeedback zu dem Bediener in bezug auf den Status und die Bedingungen der gesteuerten Prozesse **112**. Die Programmierworkstation **106** umfaßt einen Prozessor **116** sowie ein Display **115** und eine oder mehrere Ein-/Ausgabe- oder Benutzeroberflächen-Einrichtungen **118** wie etwa eine Tastatur, einen Lichtgriffel und dergleichen. Die Workstation weist ferner einen Speicher **117** auf, der sowohl einen flüchtigen als auch einen nichtflüchtigen Speicher umfaßt. Der Speicher **117** enthält ein Steuerprogramm, das an dem Prozessor **116** ausgeführt wird, um Operationen und Funktionen der Prozeßsteuerungsumgebung **100** zu implementieren. Der Speicher **117** weist ferner ein Konfigurierungs- und Verwaltungs-Programmierwerkzeug **120** auf (das auch als ein Steuerungsstudio-Programmierwerkzeug bezeichnet wird). Die Bedienerworkstation **102** und andere (nicht gezeigte) Workstations in der Prozeßsteuerungsumgebung **100** umfassen wenigstens eine Zentraleinheit (nicht gezeigt), die mit einem Display (nicht gezeigt) und einer Benutzeroberflächen-Einrichtung (nicht gezeigt) elektrisch verbunden ist, um einen Dialog zwischen einem Benutzer und dem Prozessor zu ermöglichen.

[0033] Das Programmierwerkzeug **120** dient dazu, das Prozeßsteuerungsnetz zu konfigurieren und sicherzustellen, daß das Prozeßsteuerungsnetz mit einem gewünschten Standardprotokoll, beispielsweise dem Feldbus-Protokoll, übereinstimmt. Das Programmierwerkzeug **120** kann auch bei der Verwaltung des Prozeßsteuerungsnetzes verwendet wer-

den, um eine effizientere Fehlersuche und Wartung zu ermöglichen. Das Programmierwerkzeug **120** ist bevorzugt Software, die in dem Speicher **117** gespeichert ist, kann aber auf computerlesbaren Datenträgern enthalten sein und wird von dem Prozessor **116** der Programmierwerkstation **106** ausgeführt. Der computerlesbare Datenträger kann eine Diskette, ein CD-ROM oder jede andere Art von Datenträger sein, auf dem Software gespeichert werden kann. Das Programmierwerkzeug **120** ermöglicht Bildschirmpräsentationen, die auf dem Display **115** der Programmierwerkstation **106** präsentiert werden, die entweder nur die logische Verknüpfung des Prozesses **112** zeigen oder physische Verbindungen enthalten können, die die räumlichen Charakteristiken des Layouts der Raffinerie oder Fertigungsanlage enthalten.

**[0034]** Das Programmierwerkzeug der Erfindung wird zwar im einzelnen im Zusammenhang mit einem Prozeßsteuerungssystem beschrieben, das Feldbus-Einrichtungen verwendet; es ist aber zu beachten, daß das Programmierwerkzeug der Erfindung mit Prozeßsteuerungssystemen verwendbar ist, die andere Arten von Feldeinrichtungen und Kommunikationsprotokollen aufweisen, was Protokolle, die mit anderen als Zweidrahtbussen arbeiten, und Protokolle, die nur analoge oder sowohl analoge als auch digitale Kommunikation unterstützen, einschließt. So kann das Programmierwerkzeug der Erfindung beispielsweise in jedem Prozeßsteuerungssystem verwendet werden, das unter Anwendung der Kommunikationsprotokolle HART, PROFIBUS usw. oder irgendwelcher anderer Kommunikationsprotokolle kommuniziert, die es heute gibt oder die vielleicht in Zukunft entwickelt werden.

**[0035]** Eine allgemeine Beschreibung des Feldbus-Protokolls, der nach diesem Protokoll konfigurierten Feldeinrichtungen, der Art und Weise, wie die Kommunikation in einer Prozeßsteuerungsumgebung stattfindet, die das Feldbus-Protokoll implementiert, und von beispielhaften Beschränkungen von Werten, die unter dem Feldbus-Protokoll erforderlich sind, folgt anschließend. Es versteht sich jedoch, daß das Feldbus-Protokoll im Stand der Technik bekannt und im einzelnen in zahlreichen Artikeln, Broschüren und Spezifikationen beschrieben ist, die unter anderem von der Feldbus Foundation, einer gemeinnützigen Organisation in Austin, Texas, veröffentlicht, verteilt und verfügbar gemacht werden. Insbesondere ist das Feldbus-Protokoll einschließlich der Beschränkungen von Werten, die unter dem Feldbus-Protokoll erforderlich sind, im einzelnen beschrieben in "Wiring and Installation 31.25 Kbits/sec. Voltage Mode Wire Medium Application Guide", Foundation Fieldbus, 1996.

**[0036]** Allgemein ist das Feldbus-Protokoll ein digitales, serielles, wechselseitiges Kommunikationspro-

tokoll, das eine standardisierte physische Schnittstelle zu einer Zweidrahtschleife oder einem Bus bildet, die/der Prozeßsteuerungsgeräte wie Sensoren, Betätigungselemente, Controller, Ventile usw. miteinander verbindet, die in einer Instrumenten- oder Prozeßsteuerungsumgebung vorhanden sind. Das Feldbus-Protokoll bildet tatsächlich ein lokales Datennetz für Feldeinstrumente (Feldeinrichtungen) innerhalb eines Prozesses, das es diesen Einrichtungen ermöglicht, Steuerfunktionen an Orten auszuführen, die über einen gesamten Prozeß verteilt sind, und miteinander vor und nach der Durchführung dieser Steuerfunktionen zu kommunizieren, um eine Steuerungs-Gesamtstrategie zu implementieren. Da das Feldbus-Protokoll es ermöglicht, Steuerfunktionen über ein gesamtes Prozeßsteuerungsnetz zu verteilen, verringert das Protokoll die Komplexität des zentralisierten Prozeßcontrollers oder eliminiert sogar vollständig die Notwendigkeit für einen solchen. Die verteilte Beschaffenheit des Systems bringt wiederum Komplexität bei der Verwaltung des Systems und bei der Feststellung des physischen Orts von problematischen Einrichtungen zur Fehlersuche und Verwaltung des Systems mit sich.

**[0037]** Das Feldbus-Protokoll ermöglicht das Verwalten der Feldeinrichtung und des gesamten Prozeßsteuerungssystems, indem es die Kommunikation über Einrichtungsbeschreibungen und Funktionsblöcke bereitstellt. Feldeinrichtungen sind Feldeinstrumente wie Sender und Ventile mit Prozessoren, die das Betriebsverhalten und den Zustand der Einrichtungen überwachen. Eine Einrichtungsbeschreibung ist ähnlich einem Treiber für die Einrichtung. Bei Feldeinrichtungen umfaßt die Einrichtungsbeschreibung die Kalibriervorgänge, Parameter und andere Informationen, die von dem Steuerungssystem benötigt werden, um mit der Feldeinrichtung zu kommunizieren. Feldeinrichtungen melden dem Steuerungssystem Standard-Betriebsparameter und führen Selbstdiagnosen durch und können Probleme wie etwa "Instrument nicht kalibriert" an das Steuerungssystem melden. Jede Feldeinrichtung hat ein Einrichtungsetikett und eine entsprechende Netzadresse.

**[0038]** Zur Verwaltung von Feldeinrichtungen sind viele Kommunikationsarten verfügbar, und zwar unter anderem: Gewinnung von Port- und Kommunikationsstatistiken, Erhalten des Status der Feldeinrichtung, Betrachten und Ändern der Ressourcenkonfiguration und -parameter, Auslösen einer Masterrückstellung oder eines Selbsttests der Feldeinrichtung, Anzeigen des Status der Sensoren und Ändern des oberen, unteren und Nullabgleichs der Sensoren. Durch Bereitstellen von räumlichen Informationen mit den oben aufgeführten Verwaltungskommunikationen können die Verwaltungsmerkmale des Prozeßsteuerungssystems effizienter und einfacher genutzt werden.

**[0039]** [Fig. 1b](#) zeigt den Controller/Multiplexer- und Prozeßbereich des Prozeßsteuerungsnetzes **100** von [Fig. 1a](#) entsprechend dem Fieldbus-Protokoll. Das Netz umfaßt einen Controller/Multiplexer **110**, der mit einem oder mehreren Prozessen **112**, die aus einer Vielzahl von Feldeinrichtungen bestehen, über einen Bus **142** gekoppelt ist. Der Bus **142** hat eine Vielzahl von Abschnitten oder Segmenten mit entsprechender Länge sowie anderen Charakteristiken. Der Bus **142** kann ferner ein oder mehr Verzweigungsstücke **144** (JB1, JB2, JB3) aufweisen, die häufig als "Bricks" bezeichnet werden. Jedes Verzweigungsstück **144** kann mit einer oder mehreren Feldbuseinrichtungen **146** zu dem Bus **142** gekoppelt sein. Der Controller/Multiplexer **110** ist außerdem mit wenigstens einer Energieversorgung **148** verbunden. Das in [Fig. 1](#) gezeigte Netz ist nur beispielhaft; es gibt viele andere Möglichkeiten, ein Prozeßsteuerungsnetz unter Anwendung des Fieldbus-Protokolls zu konfigurieren.

**[0040]** Das Prozeßsteuerungsnetz **100** umfaßt eine Reihe von räumlichen Charakteristiken wie etwa die Stichlänge eines bestimmten Kommunikationsabschnitts, die Gesamtlänge des Busses, die Gesamtzahl von Prozeßsteuerungseinrichtungen, die mit einem bestimmten Verzweigungsstück gekoppelt sind, und den physischen Ort der Controller und Einrichtungen in bezug auf das Layout der Raffinations- oder Herstellungsanlage. Diese räumlichen Charakteristiken können während der Konfiguration des Systems automatisch gemessen und errechnet werden unter Nutzung der räumlichen Informationen hinsichtlich des physischen Layouts der Anlage. Das Prozeßsteuerungsnetz **100** umfaßt ferner eine Reihe von nichträumlichen Charakteristiken wie den Spannungsabfall über Kommunikationsabschnitten, die Gesamtstromentnahme eines Segments und die Arten von Einrichtungen im System. Diese nichträumlichen Charakteristiken werden vom Benutzer beim Konfigurieren des Systems bereitgestellt. Das Programmierwerkzeug **120** analysiert diese Charakteristiken, um festzustellen, ob das Prozeßsteuerungsnetz dem gewünschten Standardprotokoll entspricht.

**[0041]** Nach beendeter Konfigurierung des Prozeßsteuerungssystems kann das Programmierwerkzeug **120** einschließlich des räumlichen Layouts des Systems in der Anlage genutzt werden, um das Prozeßsteuerungssystem zu verwalten, wobei eine der Workstations **102**, **104** oder **106** verwendet wird. Die Funktion der Verwaltung des Prozeßsteuerungssystems umfaßt Funktionen wie Diagnose, Online-Fehlersuche, Alarmüberwachung und Wartung der Einrichtungen. Während der Diagnose und der Alarmüberwachung kann, wenn ein Ventil oder eine andere Einrichtung ausfällt, die Darstellung der Einrichtung auf dem Bildschirm der Displayeinrichtung in der räumlichen Ansicht des Displays blinken und leicht aufgefunden werden. Der Etikettenname der Einrich-

tung sowie der räumliche Ort der Einrichtung können genutzt werden, um das Ventil oder die sonstige Einrichtung zu identifizieren. Bei der Online-Fehlersuche können die Werte der Verbinder und die Attribute in den Funktionsblöcken in der räumlichen Ansicht der Anlage gezeigt werden, so daß der Benutzer die aktuellen Zustände des Systems leichter feststellen kann. Während der Wartung von Einrichtungen können durch die Wahl einer Einrichtung in der räumlichen Ansicht die aktuellen Bedingungen und Informationen über die Einrichtung wie Stromflußrate oder neueste Wartungsaufzeichnungen erhalten werden.

**[0042]** Die Prozeßsteuerungsumgebung **100** existiert in einem Konfigurationsmodell oder einer Konfigurationsimplementierung **210** und einem Verwaltungs- oder Ausführungszeitmodell oder einer solchen Implementierung **220**, wie [Fig. 2](#) zeigt. In der Konfigurationsimplementierung **210** sind die einzelnen Einrichtungen, Objekte, Verbindungen und Beziehungen innerhalb der Prozeßsteuerungsumgebung **100** definiert und zu der räumlichen Information in bezug auf das physische Layout der Anlage in Beziehung gesetzt. In der Ausführungszeitimplementierung **220** werden Operationen der verschiedenen Einzeleinrichtungen, Objekte, Verbindungen und Beziehungen ausgeführt. Die Konfigurationsimplementierung **210** und die Ausführungszeitimplementierung **220** sind über eine ASCII-basierte Herunterladesprache miteinander verbunden. Zusätzlich zu dem Herunterladen von Definitionen führt die Herunterladesprache auch das Hinaufladen von Objekten und Objektwerten aus. Die Konfigurationsimplementierung **210** wird unter Anwendung einer Installationsprozedur aktiviert, um in der Ausführungszeitimplementierung **220** ausgeführt zu werden.

**[0043]** Die Prozeßsteuerungsumgebung **100** weist eine Vielzahl von Untersystemen auf, wobei einige der Untersysteme sowohl eine Konfigurations- als auch eine Ausführungszeitimplementierung haben. Beispielsweise liefert ein Prozeßgraphikuntersystem **230** benutzerdefinierte Ansichten und Bedieneroberflächen zu der Architektur der Prozeßsteuerungsumgebung **100**. Das Prozeßgraphikuntersystem **230** hat einen Prozeßgraphikeditor **232**, einen Teil der Konfigurationsimplementierung **210**, sowie einen Prozeßgraphikmaschinencodeleser **234**, einen Teil der Ausführungszeitimplementierung **220**. Der Prozeßgraphikeditor **232** ist mit dem Prozeßgraphikmaschinencodeleser **234** über eine Intersubsystemschnittstelle **236** in der Herunterladesprache verbunden. Die Prozeßsteuerungsumgebung **100** umfaßt ferner ein Steuerungssystem **240**, das Steuerungsmodule und Gerätemodule in einem Definitions- und Moduleditor **242** konfiguriert und installiert und die Steuerungsmodule und die Gerätemodule in einem Ausführungszeitcontroller **244** ausführt. Der Definitions- und Moduleditor **242** arbeitet innerhalb der Konfigu-

rationsimplementierung **210**, und der Ausführungszeitcontroller **244** arbeitet innerhalb der Ausführungszeitimplementierung **220**, um kontinuierliche und Ablaufsteuerungsfunktionen zu liefern. Der Definitions- und Moduleditor **242** ist mit dem Ausführungszeitcontroller **244** über eine Intersubsystemschnittstelle **246** in der Herunterladesprache verbunden. Die Vielzahl von Subsystemen ist über eine Subsystemschnittstelle **250** miteinander verbunden.

**[0044]** Die Konfigurationsimplementierung **210** und die Ausführungszeitimplementierung **220** sind an eine Masterdatenbank **260** angeschlossen, die den Zugriff auf gemeinsame Datenstrukturen unterstützt. Verschiedene lokale (Nichtmaster) Datenbanken **262** sind an die Masterdatenbank **260** angeschlossen, um beispielsweise Konfigurationsdaten von der Masterdatenbank **260** zu den lokalen Datenbanken **262** gemäß den Anweisungen eines Benutzers zu übertragen. Ein Teil der Masterdatenbank **260** ist eine Dauerdatenbank **270**. Die Dauerdatenbank **270** ist ein Objekt, das die Zeit transzendiert, so daß die Datenbank weiter besteht, nachdem der Schöpfer der Datenbank nicht mehr existiert, und den Raum transzendiert, so daß die Datenbank zu einem Adreßbereich entfernt werden kann, der von dem Adreßbereich, an dem die Datenbank geschaffen wurde, verschieden ist. Die gesamte Konfigurationsimplementierung **210** ist in der Dauerdatenbank **270** gespeichert.

**[0045]** Die Ausführungszeitimplementierung **220** ist mit der Dauerdatenbank **270** und lokalen Datenbanken **262** verbunden, um auf Datenstrukturen zuzugreifen, die durch die Konfigurationsimplementierung **210** gebildet sind. Insbesondere ruft die Ausführungszeitimplementierung **220** Gerätemodule, Displays und dergleichen von den lokalen Datenbanken **262** und der Dauerdatenbank **270** ab. Die Ausführungszeitimplementierung **220** ist mit anderen Subsystemen verbunden, um Definitionen zu installieren, wodurch Gegenstände installiert werden, die dazu dienen, Objekte zu schaffen, wenn die Definitionen noch nicht existieren, Ausführungszeitobjekte zu bilden und Informationen von verschiedenen Quellen zu Zielgegenständen zu übertragen.

**[0046]** [Fig. 3](#) ist ein Ablaufdiagramm, das die Operation des Programmierwerkzeugs **120** verdeutlicht. Die verschiedenen Schritte des Programmierwerkzeugs **120** operieren entsprechend einer "Wizard"-Funktionalität, die in verschiedenen Programmen vorhanden ist, die unter einem WINDOWS™-Betriebssystem laufen. Nach Abschluß jedes Schritts geht der Benutzer zum nächsten Schritt weiter durch Betätigen einer "NEXT"-Taste oder dergleichen. Wenn der Benutzer nicht weitergehen möchte, dann kann er aus dem Programmierwerkzeug durch Betätigen einer "EXIT"-Taste oder dergleichen aussteigen.

**[0047]** In Schritt **310** liefert der Benutzer dem Programmierwerkzeug Informationen in bezug auf die nichträumlichen Charakteristiken des Prozeßsteuerungsnetzes. Diese Informationen umfassen Dinge wie Information über den Kunden, die verwendeten Einrichtungen, Kalibrierdaten, Etikettennamen, Kabeltypen, Energieversorgungscharakteristiken und Karten-, Segment- und Übergangs-Konfigurationsinformationen. Die Kundeninformationen können umfassen: den Kundennamen, den Firmennamen, den Ort der Anlage, an der das Netz vorliegt, den Namen des Repräsentanten, der das Programmierwerkzeug bereitstellt, und den Namen einer Kontaktperson für diesen Repräsentanten. Die Kartenkonfigurationsinformation kann den Benutzer mit Informationen über die Art der verwendeten Karten und der Operationen versorgen, die zur Analyse des Prozeßsteuerungsnetzes **100** verwendet werden. Die Segmentkonfigurationsinformation kann die Spannung der Energieversorgung, den Kabeltyp (einschließlich Informationen über den Drahtdurchmesser, der in dem Kabel verwendet wird, sowie anderer Charakteristiken des Kabels) umfassen. Die Übergangsinformationen können Informationen in bezug auf die mit dem Übergang gekoppelten Einrichtungen und über die Konfiguration der Kopplung mit dem Übergang einschließlich des Sticklebels sowie über den Typ von Instrument, das mit dem Verzweigungsstück gekoppelt ist, umfassen. Bei der bevorzugten Ausführungsform ist das Instrument eine Einrichtung, die mit dem Feldbus-Protokoll übereinstimmt. Der Benutzer kann dem Instrument fakultativ eine Etikettenkennung zuordnen.

**[0048]** Zum Konfigurieren einer Karte wählt ein Benutzer eine Controllerkarte aus einer Liste von verfügbaren Controllerkarten aus. Nach der Wahl der Karte kann die relevante Information für die gewählte Controllerkarte an das Programmierwerkzeug geliefert werden. Durch die Wahl einer Controllerkarte konfiguriert der Benutzer im wesentlichen ein Segment des Netzes. Bei der bevorzugten Ausführungsform kann jede Controllerkarte zwei Segmente steuern; aber in Abhängigkeit von der Controllerkarte können von einer Controllerkarte mehr oder weniger Segmente gesteuert werden. Während die Segmente konfiguriert werden, kann der Benutzer auf eine Zusammenfassung der Informationen zugreifen, die an das Programmierwerkzeug **120** geliefert wurden.

**[0049]** In Schritt **320** liefert der Benutzer räumliche Informationen hinsichtlich der Anlage an das Programmierwerkzeug. Insbesondere werden das physische Layout der Anlage einschließlich der Etagenplangröße, des Instrumententyps, der -größe und des -orts sowie Drahtmodell Darstellungen eingegeben. Diese Informationen können von dem Benutzer in das Programmierwerkzeug eingegeben oder von einem anderen Programmierwerkzeug importiert werden, beispielsweise von 3D Toolkit, Open Inven-



tor von TGS.

**[0050]** In Schritt **330** werden Funktionsblöcke geschaffen und aktiviert. In dem Fieldbus-Protokoll liefern Funktionsblöcke die Steuerung des Systemverhaltens und können Funktionen wie Kalibriervorgänge, Parametervorgänge und Kommunikationsvorgänge umfassen. Jede Fieldbus-Einrichtung kann mehrere Funktionsblöcke haben. Die Anordnung und die Verbindungen der Blöcke bestimmen die Funktion der Fieldbus-Einrichtungen.

**[0051]** In Schritt **340** wird das physische Layout des Prozeßsteuerungssystems auf die räumliche Information hinsichtlich des Layouts der Anlage angewandt. Funktionsblöcke und Einrichtungen werden miteinander verdrahtet, wobei typischerweise Drahtmodellen und der Verdrahtung anderer Einrichtungen in der Anlage gefolgt wird. Das Layout kann manuell von dem Benutzer durchgeführt werden, oder das Programmierwerkzeug **120** kann das physische Layout des Prozeßsteuerungssystems automatisch erzeugen. Informationen wie die Länge eines Kabelsegments von einem Controller zu einem Übergang oder von einem Übergang zu einem anderen und die Länge eines StICKkabels können aus dem räumlichen Layout der Raffinations- oder Fertigungsanlage automatisch erzeugt und errechnet werden. Bei einer anderen Ausführungsform kann die Verbindung zwischen den Funktionsblöcken und Einrichtungen zuerst logisch hergestellt und zu einem späteren Zeitpunkt auf die räumliche Information hinsichtlich der Anlage angewandt werden.

**[0052]** In Schritt **350** wird die Konfiguration des Prozeßsteuerungssystems auf Übereinstimmung mit den Erfordernissen des gewählten Protokolls geprüft. Sämtliche StICKkabelängen eines Segments werden geprüft, um sicherzustellen, daß die StICKkabelängen eine vorbestimmte StICHLänge gemäß der Definition durch das Standardprotokoll nicht überschreiten. Die StICHLängen sind durch die Zahl der Instrumente an dem Segment (pro Segment) begrenzt. Das heißt, je geringer die Zahl der Instrumente, um so länger ist die zulässige StICHLänge pro Segment. Die Zahl von Einrichtungen pro Segment wird ebenfalls überprüft, um sicherzustellen, daß die Zahl von Einrichtungen eine vorbestimmte Zahl von Einrichtungen nicht überschreitet. Die zulässige Zahl von Einrichtungen kann in Abhängigkeit von dem Controller, der von dem Prozeßsteuerungsnetz verwendet wird, verschieden sein. Bei der bevorzugten Ausführungsform erlaubt der Controller das Ankoppeln von 16 Einrichtungen per Segment an den Bus. Der derzeitige Fieldbus-Standard erlaubt jedoch das Ankoppeln von bis zu 32 Einrichtungen per Segment an den Bus. Die Gesamtstromentnahme pro Segment wird überprüft, um sicherzustellen, daß die Stromentnahme die von dem Standardprotokoll zugelassene maximale Stromentnahme nicht überschreitet. Bei der bevorzugten

Ausführungsform ist die zulässige maximale Stromentnahme 376 mA/Segment. Die Gesamtsegmentkabellänge (einschließlich der StICKkabellänge) wird überprüft, um sicherzustellen, daß die Länge die von dem Standardprotokoll zugelassene maximale Segmentlänge nicht überschreitet. Bei der bevorzugten Ausführungsform ist die zulässige maximale Segmentlänge 1900 m oder 6232 feet. Die Minimumspannung pro Segment wird überprüft, um sicherzustellen, daß die Spannung an jeder mit dem Prozeßsteuerungsnetz gekoppelten Einrichtung höher oder gleich wie die Spannung ist, die von dem Standardprotokoll angegeben ist. Bei der bevorzugten Ausführungsform ist diese Spannung 12,5V. Wenn einer oder mehrere der Werte nicht innerhalb der von dem Protokoll definierten Grenzen liegen, kann der Benutzer zu Schritt **340** zurückspringen, um die Konfiguration des Prozeßsteuerungsnetzes nachzuprüfen.

**[0053]** Wenn das Prozeßsteuerungsnetz konfiguriert ist, kann der Benutzer damit beginnen, das Prozeßsteuerungssystem zu verwalten (Schritt **360**), indem er die gelieferten nicht-räumlichen und räumlichen Informationen nutzt. Zur Verwaltung der Feldeinrichtung sind viele Kommunikationsarten verfügbar, die folgende einschließen: Erhalt von Port- und Kommunikationsstatistiken, Erhalt des Status der Feldeinrichtung, Betrachten und Ändern von Ressourcenkonfiguration und -parametern, Auslösen einer Masterrückstellung oder einer Selbstprüfung der Feldeinrichtung, Anzeigen des Status der Sensoren und Ändern des oberen, unteren und Nullabgleichs der Sensoren. Durch die Bereitstellung von räumlichen Informationen zusammen mit den oben angegebenen Verwaltungskommunikationen können die Verwaltungsmerkmale des Prozeßsteuerungssystems effizienter und einfacher genutzt werden.

**[0054]** Die räumlichen Informationen in bezug auf die Anlage können vollkommen dreidimensional sein, was dreidimensionale Wände, Einrichtungen, Workstations usw. umfaßt. Die räumlichen Informationen hinsichtlich der Anlage können auch ein zweidimensionaler Plan der Anlage sein, auf den die Konfiguration des Prozeßsteuerungssystems übertragen ist, oder jede Kombination von zwei und drei Dimensionen, die für die Benutzeranwendung geeignet ist.

**[0055]** Bei anderen Ausführungsformen kann das Programmierwerkzeug dem Benutzer eine Möglichkeit bieten, eine Stückliste für die Konstruktion des Prozeßsteuerungsnetzes zu erhalten. Das Programmierwerkzeug kann außerdem das Layout des Prozeßsteuerungssystems innerhalb des physischen Layouts der Anlage automatisch bereitstellen und sicherstellen, daß den Forderungen des Protokolls entsprochen wird.

**[0056]** Bei einer anderen Ausführungsform kann der

Benutzer das System konfigurieren, ohne die räumliche Information der Anlage bereitzustellen, und kann zu einem späteren Zeitpunkt die räumliche Information hinzufügen, die bei der Verwaltung des Prozeßsteuerungssystems genutzt wird.

**[0057]** Es ist ersichtlich, daß zwar Funktionen in einer bestimmten Reihenfolge von Ereignissen beschrieben werden, jede andere Reihenfolge, in der die Information angegeben oder die Schritte vervollständigt werden, jedoch im Rahmen der Erfindung liegt.

**[0058]** [Fig. 4](#) zeigt eine Bildschirmdarstellung des Konfigurationsteils des Programmierwerkzeugs unter Verwendung der logischen Verbindungen des Prozeßsteuerungssystems in dem Hauptsteuerfenster des Programmierwerkzeugs **120**. Die Bildschirmdarstellung des Programmierwerkzeugs **120** umfaßt Text-Balkenmenüs **402**, ein Piktogramm-Menü **404**, eine Schablonenbereichsdarstellung **406** und eine Diagrammbereich-Bildschirmdarstellung **408**. Schablonen **420** sind in der Schablonenbereichsdarstellung **406** gezeigt. Das Benutzerdiagramm der Konstruktion der Prozeßsteuerungsumgebung ist in der Diagrammbereich-Bildschirmdarstellung **408** gezeigt. Dieses Diagramm der Konstruktion der Prozeßsteuerungsumgebung wird als die Prozeßsteuerungsumgebungsansicht bezeichnet. Jede der Darstellungen in dem Hauptfenster kann vom Benutzer in bezug auf Größe und Ort entsprechend den bekannten Fensertertechniken verändert werden. Das Programmierwerkzeug **120** verfolgt Ort und Größe der Ausschnitte des Hauptfensters durch Aufrechterhalten von dauerhaften Gegenstandsdaten einschließlich Koordinaten innerhalb der zweidimensionalen Anzeige sowie von Stil- und anderen Informationen.

**[0059]** Bei der Konstruktion einer Prozeßsteuerungsumgebung unter Anwendung logischer Verknüpfungen aktiviert ein Benutzer einfach eine Schablone aus der Schablonenbereichsdarstellung **408**, zieht die aktivierte Schablone zu einer gewünschten Stelle innerhalb der Diagrammbereichs-Bildschirmdarstellung **408** und läßt die aktivierte Schablone an einer gewünschten Stelle los. Ein Steuerungsstudio-Gegenstandssystem **130** erzeugt dann ein Diagrammelement, das es dem Diagramm erlaubt, einen Gegenstand mit sämtlichen Informationen, die zur Konfigurierung einer Prozeßsteuerungsumgebung erforderlich sind, zu erzeugen. Da die Schablonenelemente Gegenstände sind, die die gesamte erforderliche Information aufweisen, damit das Diagramm eine Prozeßsteuerungsumgebung konfigurieren kann, kann dann, wenn die Konstruktion der Prozeßsteuerungsumgebung innerhalb des Diagrammbereichs vollständig ist, diese Konstruktion direkt zu den entsprechenden Bereichen der Prozeßsteuerungsumgebung heruntergeladen werden.

**[0060]** Die [Fig. 4](#) und [Fig. 5a-Fig. 5c](#) zeigen Bildschirmpräsentationen des räumlichen Layoutbereichs des Programmierwerkzeugs unter Nutzung von räumlichen Informationen der Anlage in dem Layout des Prozeßsteuerungssystems. Das Programmierwerkzeug ermöglicht eine Betrachtung des räumlichen Layouts des Prozeßsteuerungssystems unter verschiedenen Winkeln und mit unterschiedlicher Vergrößerung. Die Präsentation kann eine Graustufen- oder eine Farbpräsentation sein. Die Bildschirmpräsentationen können innerhalb eines Fensters des Programmierwerkzeugs **120** analog dem Diagrammbereich-Bildschirmpräsentationsfenster **408** enthalten sein. Andere Möglichkeiten der Darstellung der räumlichen Information liegen im Rahmen der Erfindung.

**[0061]** Beim Entwerfen einer Prozeßsteuerungsumgebung unter Nutzung der räumlichen Informationen der Anlage beginnt ein Benutzer damit, daß er entweder das physische Layout der Anlage importiert oder das Layout in dem Diagrammbereich des Hauptsteuerfensters des Programmierwerkzeugs **120** erzeugt. Zum Hinzufügen von Feldeinrichtungen oder Funktionen aktiviert ein Benutzer einfach ein Schablonenelement aus der Schablonenbereichsdarstellung **408**, zieht das aktivierte Schablonenelement zu einer gewünschten Stelle in der räumlichen Darstellung der Anlage innerhalb der Diagrammbereich-Bildschirmdarstellung **408** und läßt das aktivierte Schablonenelement an einer gewünschten Stelle los. Schablonenelemente umfassen Viereckdarstellungen von Funktionen ebenso wie dreidimensionale Darstellungen von Elementen, die in einer Raffinations- oder Herstellungsanlage vorkommen, etwa Ventile, Pumpen, Tanks, Rohrleitungen usw. Ein räumlicher Bereich des Steuerungsstudio-Gegenstandssystems **130** erzeugt dann ein Diagrammelement mit den Informationen, die zur Konfigurierung einer Prozeßsteuerungsumgebung innerhalb des räumlichen Layouts einer Anlage erforderlich sind. Da die Schablonenelemente Gegenstände sind, die die gesamte erforderliche Information enthalten, so daß das Diagramm eine Prozeßsteuerungsumgebung innerhalb des räumlichen Layouts einer Anlage konfigurieren kann, kann dann, wenn der Entwurf der Prozeßsteuerungsumgebung innerhalb des Diagrammbereichs fertiggestellt ist, dieser Entwurf direkt zu den entsprechenden Bereichen der Prozeßsteuerungsumgebung einschließlich des räumlichen Bereichs des Steuerungsstudio-Gegenstandssystems heruntergeladen werden.

**[0062]** Die [Fig. 5a](#) bis [Fig. 5c](#) zeigen Beispiele einer räumlichen Bildschirmpräsentation **500** einschließlich eines Beispiels des physischen Layouts der Anlage in einer räumlichen Ansicht. Insbesondere zeigt [Fig. 5a](#) eine Präsentation eines physischen Layouts einer Anlage über einer schematischen Ansicht der Anlage: Die räumliche Präsentation umfaßt ferner

eine physische und logische Darstellung der verschiedenen Komponenten der Prozeßsteuerungs-umgebung. Somit kann ein Benutzer vorteilhaft die physischen Orte der verschiedenen Komponenten der Prozeßsteuerungs-umgebung, die einer schematischen Ansicht der Anlage überlagert sind, betrachten. Die [Fig. 5b](#) und [Fig. 5c](#) zeigen eine vergrößerte und gedrehte Ansicht von Bereichen der Diagramm-präsentation der [Fig. 5a](#). Die [Fig. 5b](#) bis [Fig. 5c](#) zeigen also Beispiele, wie ein Benutzer auf Bereiche der Diagramm-präsentation, wie sie etwa in [Fig. 5a](#) gezeigt ist, zugreifen kann, um eine bessere Ansicht von bestimmten Bereichen der Prozeßsteuerungs-umgebung zu erhalten. Es versteht sich, daß die räumliche Präsentation der schematischen Ansicht nicht notwendigerweise überlagert sein muß.

**[0063]** Andere Ausführungsformen liegen im Rahmen der nachfolgenden Ansprüche.

**[0064]** Beispielsweise analysiert das Protokoll, in dem die bevorzugte Ausführungsform beschrieben ist, zwar ein Prozeßsteuerungsnetz für ein Field-bus-Protokoll, es versteht sich aber, daß durch Justieren der jeweils geeigneten Einschränkungen jedes Protokoll analysiert werden kann.

**[0065]** Beispielsweise arbeitet zwar die bevorzugte Ausführungsform unter einem WINDOWS-Betriebssystem und verwendet eine Präsentation vom Wizard-Typ, aber es versteht sich, daß diese Einzelheiten das Gesamtkonzept der Erfindung nicht einschränken sollen.

**[0066]** Ferner wurden zwar spezielle Ausführungsformen der Erfindung gezeigt und beschrieben, für den Fachmann ist jedoch ersichtlich, daß Änderungen und Modifikationen vorgenommen werden können, ohne vom Umfang der Erfindung abzuweichen; daher sollen die beigefügten Ansprüche alle derartigen Änderungen und Modifikationen mit umfassen, die unter den eigentlichen Umfang der Erfindung fallen, was auch Implementierungen in anderen Programmiersprachen umfaßt, jedoch nicht darauf beschränkt ist. Außerdem ist die bevorzugte Ausführungsform zwar als eine Softwareimplementierung beschrieben, es versteht sich jedoch, daß Hardwareimplementierungen wie etwa anwenderspezifische Implementierungen mit integrierten Schaltungen ebenfalls im Rahmen der nachfolgenden Ansprüche liegen.

### Patentansprüche

1. Verfahren zum Konfigurieren und Verwalten eines Prozeßsteuerungsnetzes, wobei das Prozeßsteuerungsnetz einen Computer mit einem Prozessor und einem Speicher aufweist, wobei das Verfahren die folgenden Schritte aufweist:  
Versorgen des Computers mit auf ein räumliches

Layout einer Anlage bezogenen Informationen;  
Versorgen des Computers mit auf die Konfiguration des Prozeßsteuerungsnetzes bezogenen Informationen;  
Präsentieren eines auf das räumliche Layout der Anlage bezogenen Layouts der Konfiguration des Prozeßsteuerungsnetzes,  
gekennzeichnet durch  
ein Analysieren des Layouts des Prozeßsteuerungsnetzes unter Bezugnahme auf das physische Layout der Anlage, um sicherzustellen, daß das Layout des Prozeßsteuerungsnetzes mit den Kriterien eines Standardprotokolls für Prozeßsteuerungsnetzwerke übereinstimmt, welches nicht auf der Basis von TCP/IP beruht.

2. Verfahren nach Anspruch 1, gekennzeichnet durch Verwalten des Prozeßsteuerungssystems in einer Ausführungszeitumgebung.

3. Verfahren nach Anspruch 1, dadurch gekennzeichnet, daß das Bereitstellen eines Layouts des Prozeßsteuerungsnetzes das automatische Erzeugen des Layouts umfaßt.

4. Verfahren nach Anspruch 2, dadurch gekennzeichnet, daß das Verwalten des Prozeßsteuerungssystems das Vorsehen einer blinkenden Darstellung einer Einrichtung in einer räumlichen Ansicht des Prozeßsteuerungssystems umfaßt, um aktive Alarmer zu bezeichnen.

5. Vorrichtung zum Konfigurieren und Verwalten eines Prozeßsteuerungsnetzes, welche Folgendes aufweist:  
einen Computer, der einen Prozessor (**116**) und einen Speicher (**117**) hat;  
ein Programmierwerkzeug (**120**), das an dem Computer ausführbar ist;  
Mittel zum Versorgen des Computers mit auf ein räumliches Layout einer Anlage bezogenen Informationen;  
Mittel zum Versorgen des Computers mit Informationen, die sich auf Materialien beziehen, die in dem Prozeßsteuerungsnetz verwendet werden;  
wobei das Programmierwerkzeug (**120**) verwendet wird, um ein auf das räumliche Layout der Anlage angewandtes Layout des Prozeßsteuerungsnetzes bereitzustellen,  
dadurch gekennzeichnet, daß  
das Programmierwerkzeug (**120**) verwendet wird, um das auf das räumliche Layout der Anlage angewandte Layout des Prozeßsteuerungsnetzes zu analysieren, um sicherzustellen, daß das Layout des Prozeßsteuerungsnetzes mit Kriterien eines Standardprotokolls für Prozeßsteuerungsnetzwerke übereinstimmt, welches nicht auf der Basis von TCP/IP beruht.

6. Vorrichtung nach Anspruch 5, dadurch gekennzeichnet, daß das Programmierwerkzeug (**120**)

angewandt wird, um das Prozeßsteuerungssystem in einer Ausführungszeitumgebung zu verwalten.

7. Vorrichtung nach Anspruch 5, dadurch gekennzeichnet, daß das Programmierwerkzeug (**120**), das zum Bereitstellen eines Layouts des Prozeßsteuerungsnetzes genutzt wird, ferner das automatische Erzeugen des Layouts durch das Programmierwerkzeug aufweist.

8. Vorrichtung nach Anspruch 6, dadurch gekennzeichnet, daß das Programmierwerkzeug (**120**), das zum Verwalten des Prozeßsteuerungssystems genutzt wird, eine blinkende Darstellung einer Einrichtung in einer räumlichen Ansicht des Prozeßsteuerungssystems aufweist, um aktive Alarme anzuzeigen.

9. Verfahren zum Konfigurieren und Verwalten eines Prozeßsteuerungsnetzes, das die folgenden Schritte aufweist:

Vorsehen eines an einem Computer ausgeführten Programmierwerkzeugs;

Versorgen des Computers mit Informationen, die in dem Prozeßsteuerungsnetz verwendete Materialien betreffen;

Nutzen des Programmierwerkzeugs, um ein Layout des Prozeßsteuerungsnetzes zu erstellen; und

Versorgen des Computers zu einem späteren Zeitpunkt mit Informationen, die ein räumliches Layout einer Anlage betreffen;

Anwenden des Layouts des Prozeßsteuerungsnetzes auf das räumliche Layout der Anlage, gekennzeichnet durch

ein Analysieren des Layouts des Prozeßsteuerungsnetzes unter Bezugnahme auf das physische Layout der Anlage, um sicherzustellen, daß das Layout des Prozeßsteuerungsnetzes mit den Kriterien eines Standardprotokolls für Prozeßsteuerungsnetzwerke übereinstimmt, welches nicht auf der Basis von TCP/IP beruht.

10. Verfahren nach Anspruch 1, wobei das Versorgen mit dem Layout des Prozeßsteuerungsnetzwerks den Schritt aufweist, das Layout des Prozeßsteuerungsnetzwerkes über einer schematischen Ansicht der Anlage darzustellen.

11. Vorrichtung nach Anspruch 6, wobei das Programmierwerkzeug verwendet wird, um das Layout des Prozeßsteuerungsnetzwerkes über einer schematischen Ansicht der Anlage darzustellen.

Es folgen 8 Blatt Zeichnungen

Anhängende Zeichnungen

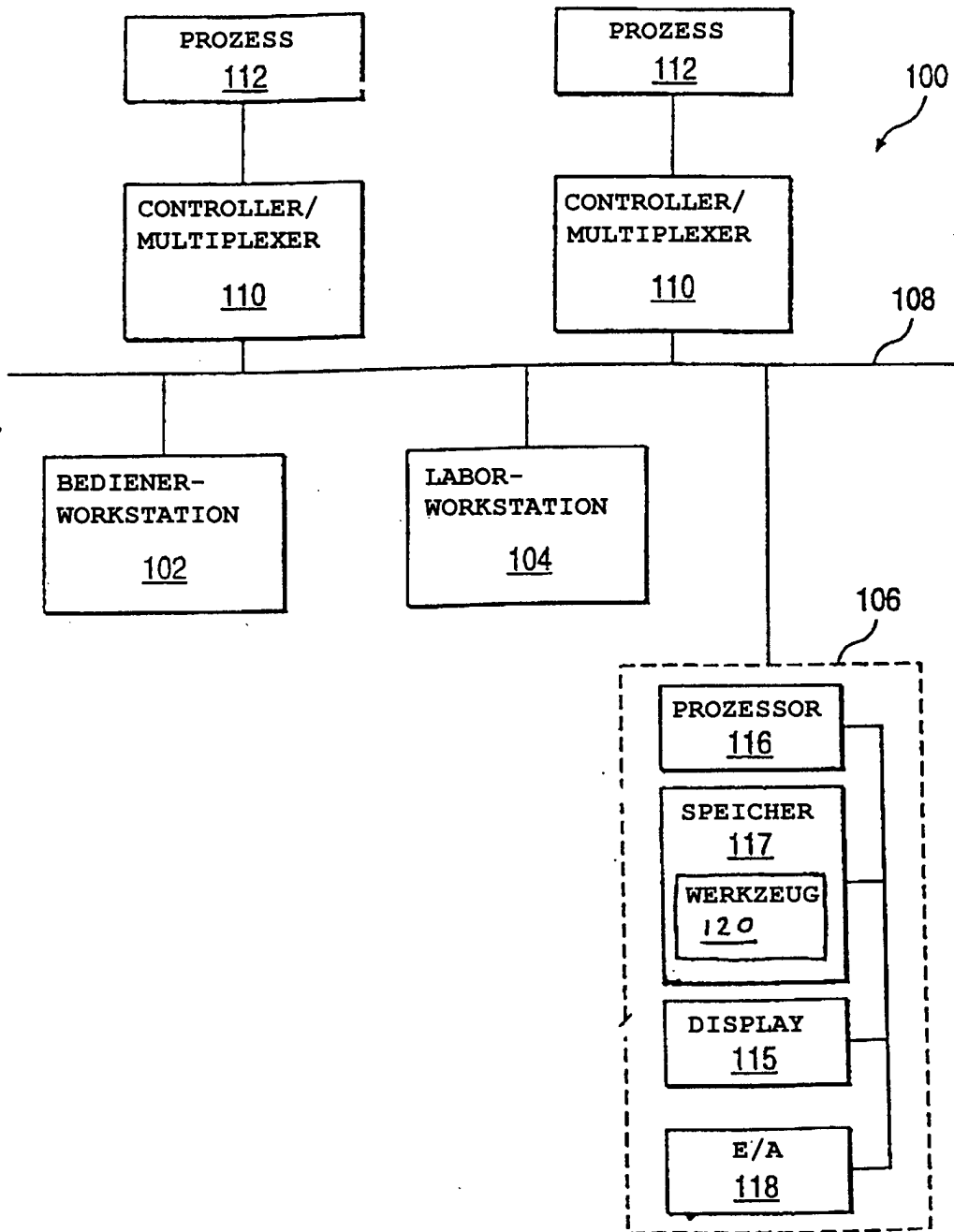
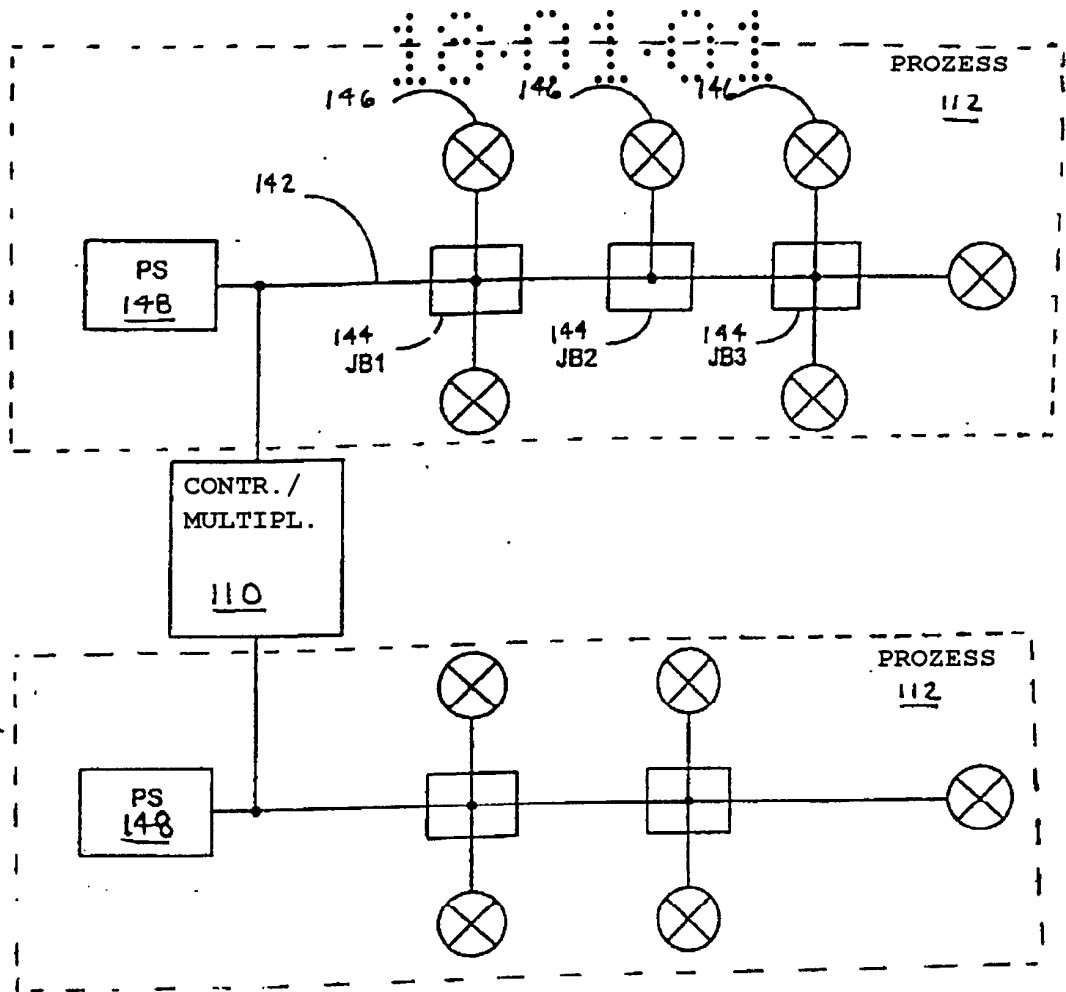


FIG. 1a



Figur 1b

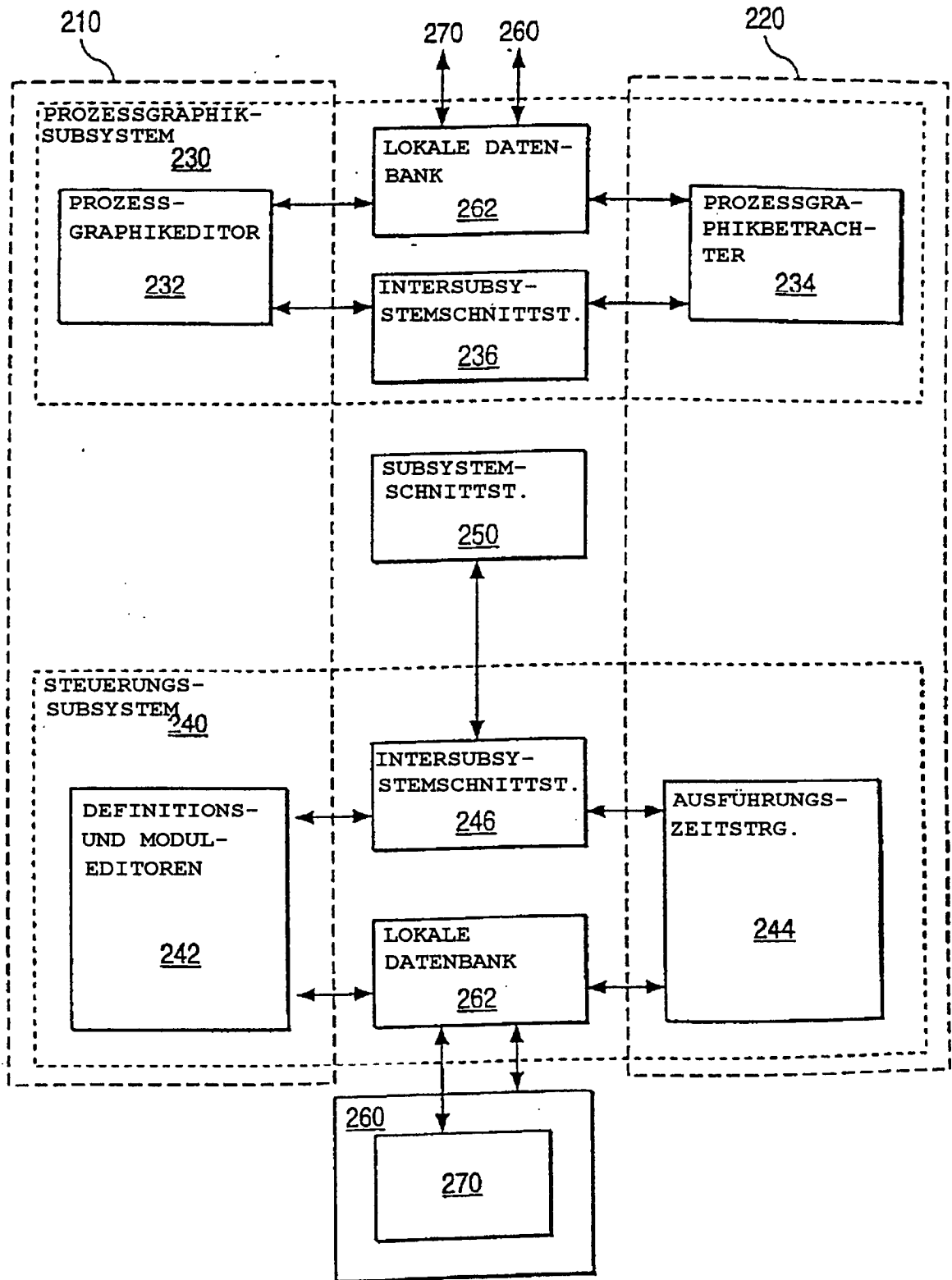
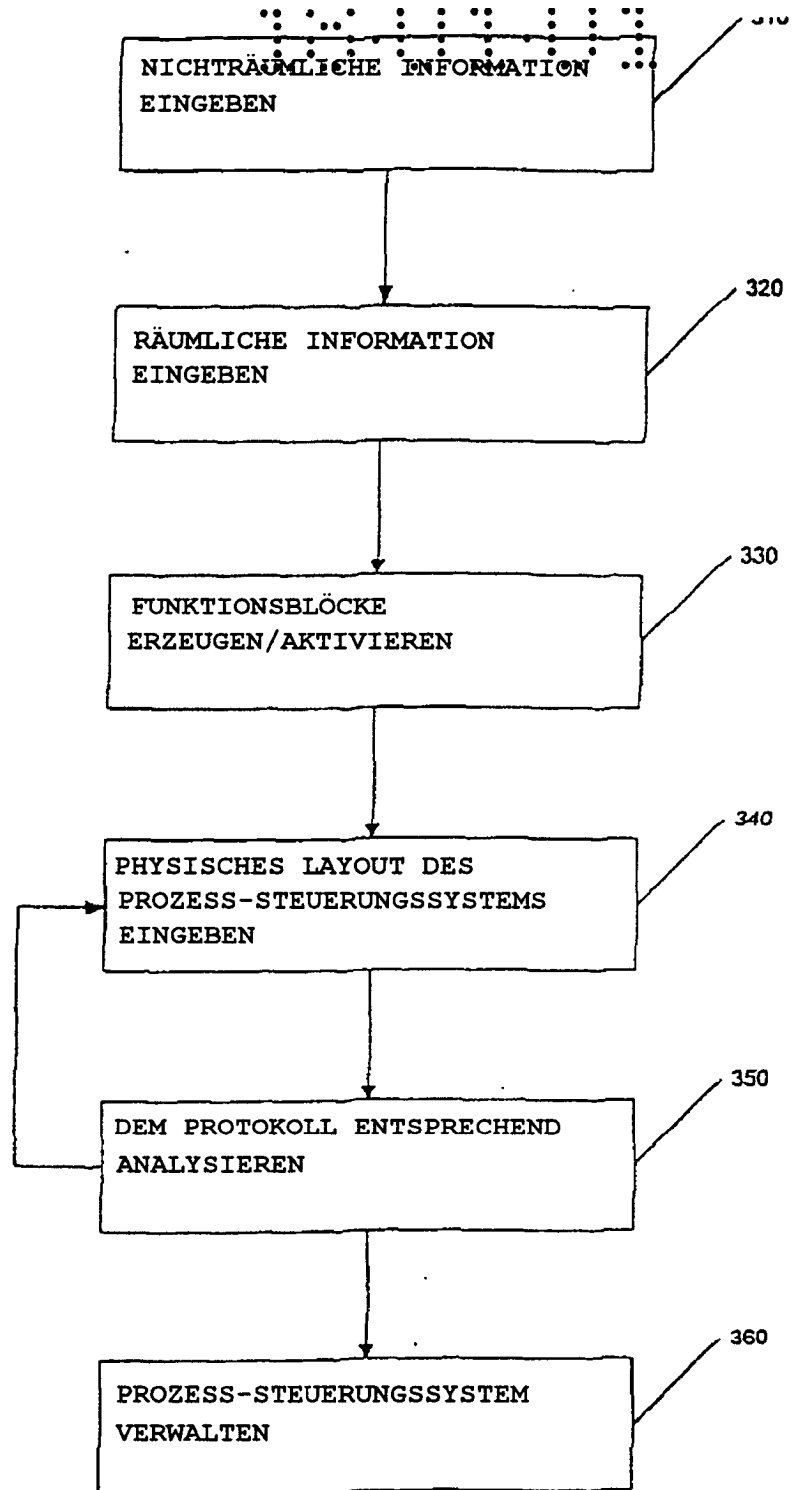


FIG. 2



FIGUR 3



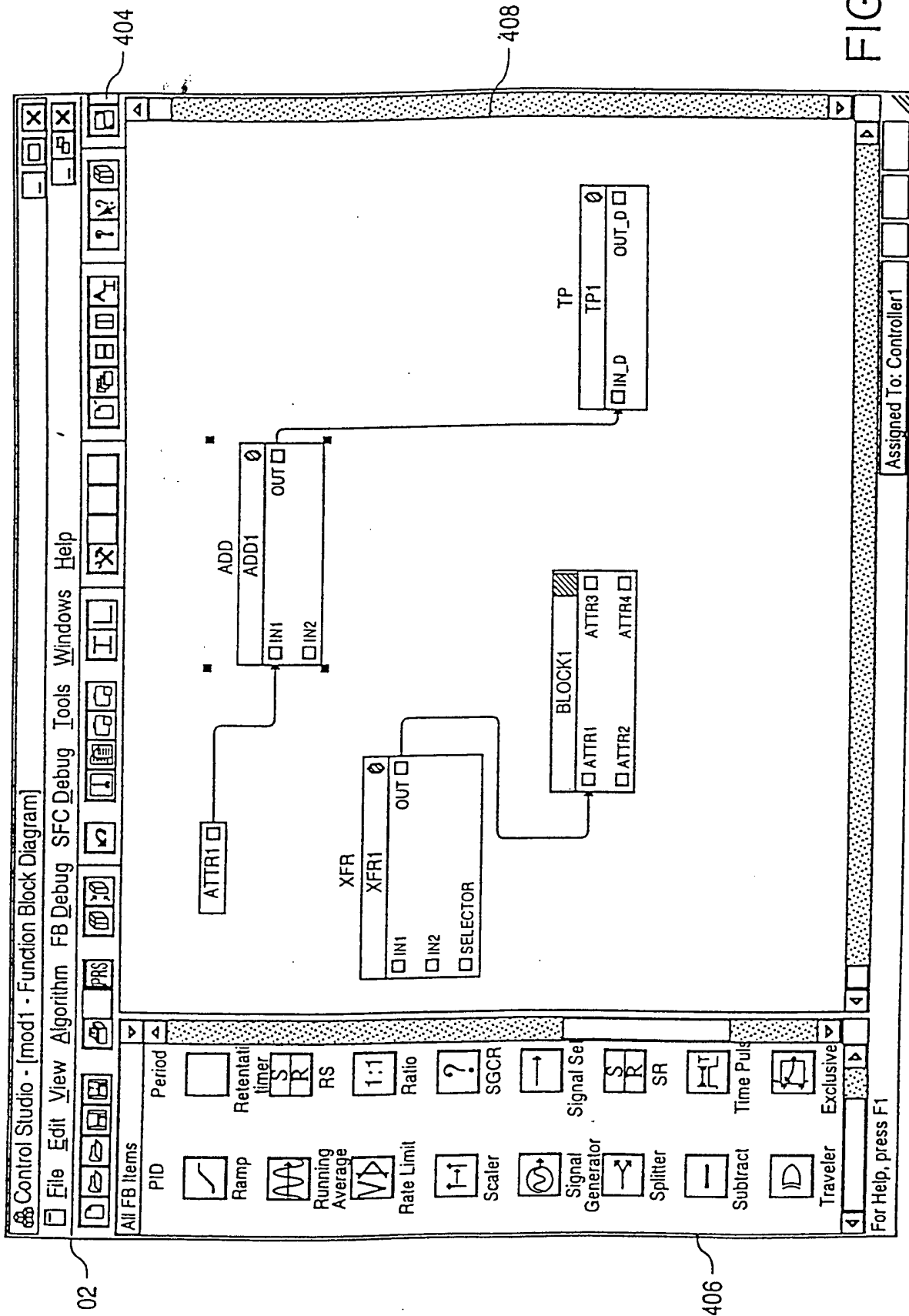


FIG. 4

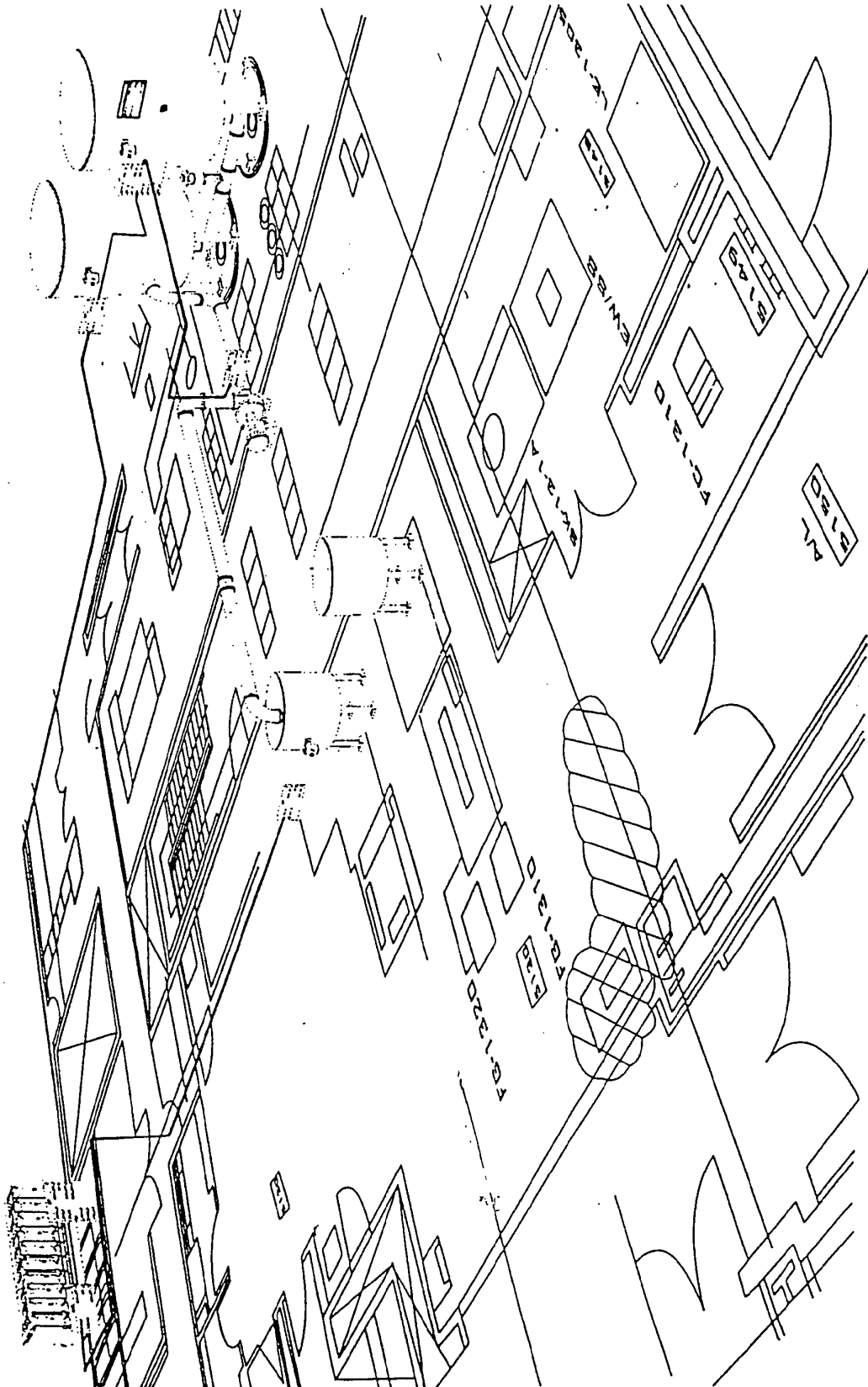


FIG. 5a

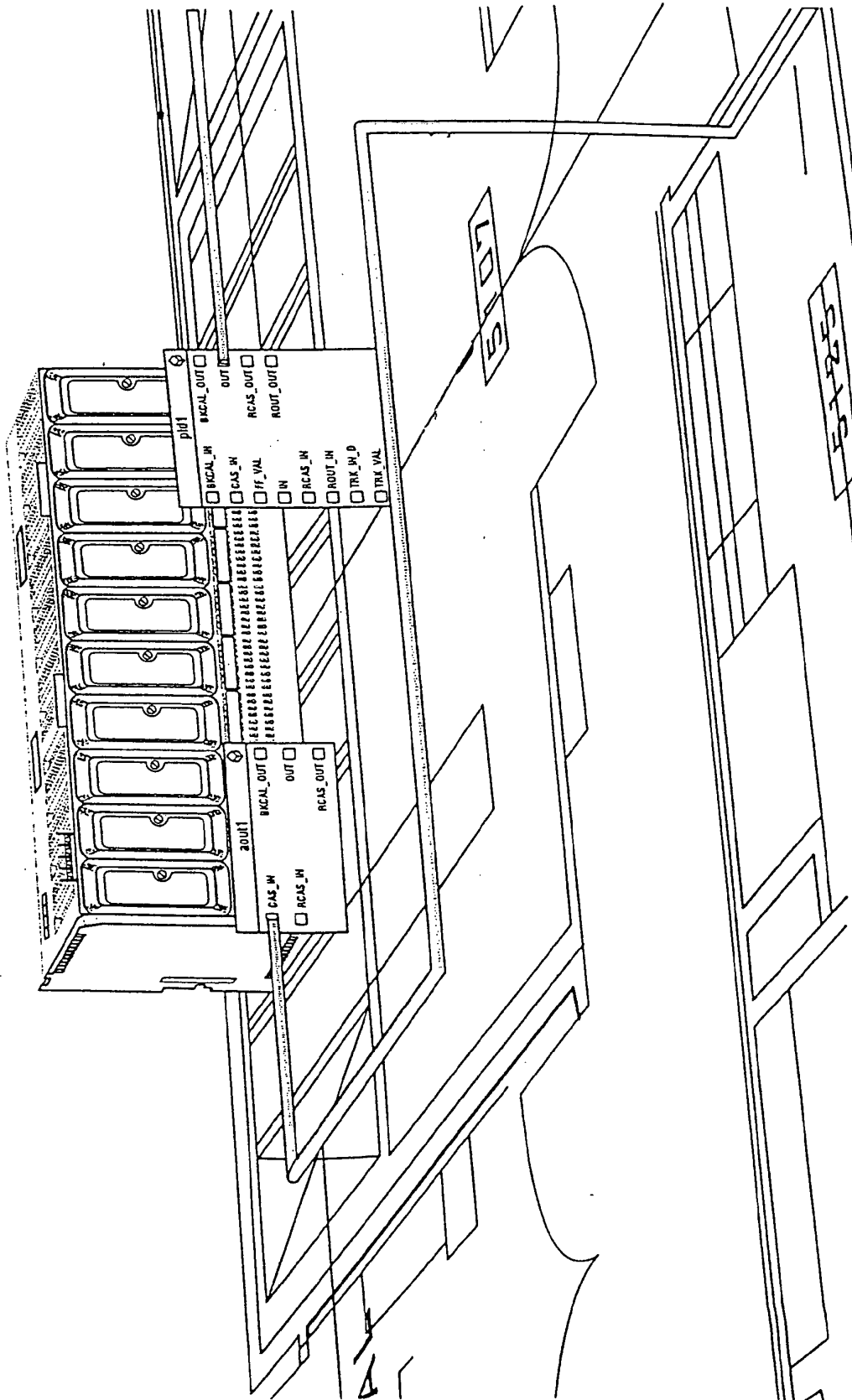


FIG. 5b

