



(10) **DE 10 2011 004 363 B4** 2023.10.05

(12)

Patentschrift

(21) Aktenzeichen: **10 2011 004 363.2**
(22) Anmeldetag: **18.02.2011**
(43) Offenlegungstag: **23.08.2012**
(45) Veröffentlichungstag
der Patenterteilung: **05.10.2023**

(51) Int Cl.: **H04L 41/00 (2022.01)**
G06F 13/42 (2006.01)
H04L 12/403 (2006.01)
H04L 41/0806 (2022.01)
H04L 41/0893 (2022.01)

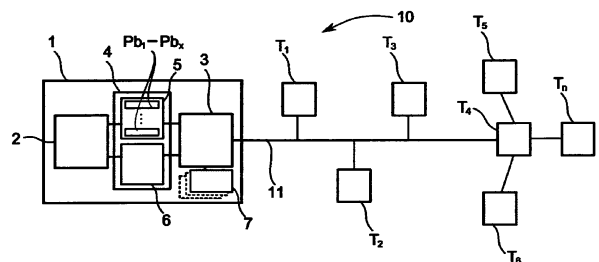
Innerhalb von neun Monaten nach Veröffentlichung der Patenterteilung kann nach § 59 Patentgesetz gegen das Patent Einspruch erhoben werden. Der Einspruch ist schriftlich zu erklären und zu begründen. Innerhalb der Einspruchsfrist ist eine Einspruchsgebühr in Höhe von 200 Euro zu entrichten (§ 6 Patentkostengesetz in Verbindung mit der Anlage zu § 2 Abs. 1 Patentkostengesetz).

| | | | | | | | |
|---|---|----|------------------|----|----|------------------|----|
| <p>(73) Patentinhaber: Airbus Operations GmbH, 21129 Hamburg, DE</p> <p>(74) Vertreter: isarpate - Patent- und Rechtsanwälte Barth Hassa Peckmann und Partner mbB, 80801 München, DE</p> <p>(72) Erfinder: Kliem, Daniel, 21075 Hamburg, DE; Wagner, Martin, 22761 Hamburg, DE</p> | <p>(56) Ermittelter Stand der Technik:</p> <table><tr><td>US</td><td>2002 / 0 112 070</td><td>A1</td></tr><tr><td>US</td><td>2008 / 0 189 459</td><td>A1</td></tr></table> <p>CENA, G. [et al.]: Efficient Polling of Devices in CANOpen Networks, Published in: Emerging Technologies and Factory Automation, 2003. Proceedings. ETFA '03. IEEE Conference (Volume:1), Date of Conference: 16-19 Sept. 2003, Page(s):123 - 130, URL: http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=1247697 [abgerufen im Internet am 28.08.2015]</p> | US | 2002 / 0 112 070 | A1 | US | 2008 / 0 189 459 | A1 |
| US | 2002 / 0 112 070 | A1 | | | | | |
| US | 2008 / 0 189 459 | A1 | | | | | |

(54) Bezeichnung: **Steuervorrichtung zum Steuern von Netzwerkteilnehmern, Verfahren zum Betreiben eines Computernetzwerks und Computernetzwerk**

(57) Hauptanspruch: Steuervorrichtung (1) zum Steuern einer Vielzahl von Netzwerkteilnehmern ($T_1 - T_n$) eines Computernetzwerks (10), mit einer Master-Recheneinrichtung (2) zum Steuern der Netzwerkteilnehmer ($T_1 - T_n$) entsprechend einem Programm, welche jedem Netzwerkteilnehmer ($T_1 - T_n$) mindestens ein Programm ($P_1 - P_m$) zuweist, mit einer Kontrolllogik (3), welche dazu ausgelegt ist, die Netzwerkteilnehmer ($T_1 - T_n$) entsprechend den diesen Netzwerkteilnehmern ($T_1 - T_n$) jeweils zugewiesenen Programmen ($P_1 - P_m$) zu gruppieren und denjenigen Netzwerkteilnehmern ($T_1 - T_n$) derselben Gruppe, denen von der Master-Recheneinrichtung (2) dasselbe Programm ($P_1 - P_m$) zugewiesen ist, dieses Programm ($P_1 - P_m$) parallel zu übermitteln, wobei ein Speicher (4) vorgesehen ist, in welchem die den Netzwerkteilnehmern ($T_1 - T_n$) zugeordneten Programme ($P_1 - P_m$), eine Gruppeninformation der Zuordnung der Programme ($P_1 - P_m$) zu den jeweiligen Netzwerkteilnehmern ($T_1 - T_n$) und eine Längeninformation über die jeweilige Programmlänge abgelegt sind; und wobei der Speicher (4) einen Programm-Speicher (5) aufweist, der dazu ausgebildet ist, die den Netzwerkteilnehmern ($T_1 - T_n$) zugewiesenen Programme $P_1 - P_m$ zu speichern, und dass der Speicher (4) einen Zuordnungs-Speicher (6) aufweist, der dazu ausgebildet ist, die mindestens eine Zuord-

nung der Netzwerkteilnehmer ($T_1 - T_n$) zu einer Gruppe von Netzwerkteilnehmern ($T_1 - T_n$) und die Programmlängen der diesen jeweils zugewiesenen Programme ($P_1 - P_m$) zu speichern; wobei die Master-Recheneinrichtung (2) ausgebildet ist, neue Programme ($P_1 - P_m$) und Zuordnung der Programme ($P_1 - P_m$) zu erzeugen und in dem Speicher (4) abzulegen, falls die vorherigen Programme ($P_1 - P_m$) und Zuordnung der Programme ($P_1 - P_m$) geändert werden sollen.



Beschreibung

Gebiet der Erfindung

[0001] Die Erfindung betrifft eine Steuervorrichtung zum Steuern einer Vielzahl von Netzwerkteilnehmern eines Computernetzwerks. Die Erfindung betrifft ferner ein Verfahren zum Betreiben eines Computernetzwerks sowie ein Computernetzwerk.

Technischer Hintergrund

[0002] Die vorliegende Erfindung befindet sich im Umfeld der vernetzten Systeme, wie etwa Computernetzwerke oder Bussysteme. Ein Computernetzwerk ist ein Zusammenschluss verschiedener technischer, primär selbstständiger elektronischer Systeme, wie etwa Computer, Sensoren, Aktoren, funktionaler Komponenten, etc., nachfolgend kurz als Teilnehmer oder Netzwerkteilnehmer bezeichnet, wobei hier vor allem die Möglichkeit zur zentralen Verwaltung von Teilnehmern des Computernetzwerks von Bedeutung ist.

[0003] Eine übliche und vorteilhafte Architektur zur zentral organisierten Verwaltung der Teilnehmer ist die so genannte Master-Slave-Architektur, die eine Form der hierarchischen Verwaltung des Zugriffs auf eine gemeinsame Ressource, wie etwa einen Speicher, darstellt. Bei einem in einer Master-Slave-Architektur aufgebauten Computernetzwerk fungiert ein Teilnehmer als Master, alle anderen Teilnehmer sind die Slaves. Der Master hat als einziger das Recht, unaufgefordert auf die gemeinsame Ressource zuzugreifen. Diese Master fungieren damit als Steuervorrichtungen zum Steuern der als Slaves arbeitenden anderen Netzwerkteilnehmer desselben Computernetzwerks. Die vorliegende Erfindung sowie die ihr zugrunde liegende Problematik wird nachfolgend mit Bezug auf ein Master-Slave Computernetzwerk in einem Flugzeug erläutert, ohne die Erfindung dahingehend einzuschränken.

[0004] Um in einem Master-Slave Computernetzwerk die Slaves zu steuern, generiert der Master für alle von ihm zu steuernden Slaves einzeln Steuerbefehle und übermittelt diese Steuerbefehle üblicherweise in einem festen Zeitraster an die entsprechenden Slaves. In den meisten Master-Slave Computernetzwerken steuert ein einzelner Master alle vorhandenen Slaves. Ein solches Netzwerk wird auch als Single-Master Netzwerk bezeichnet.

[0005] Neben diesen Steuerungsaufgaben zum Steuern der Slaves führt ein Master zusätzlich meist weitere Rechenaufgaben aus, beispielsweise das Einlesen, Abspeichern und Überwachen von Sensordaten, das Erzeugen von Echtzeitdaten, etc. Die Rechenbelastung des Masters und damit die Funktionsfähigkeit des Master-Slave Computernetzwerks

hängen wesentlich von der Rechenleistung des Masters, dessen Rechen- und Steueraufgaben sowie der Anzahl der Slaves, welche durch den Master gesteuert werden müssen, ab. Bei einer geringen Anzahl von Slaves ist dies unkritisch. Problematisch ist aber, wenn das Master-Slave Computernetzwerk eine Vielzahl von 1000 und mehr Slaves aufweist, welche jeweils durch denselben Master gesteuert werden. Solche Computernetzwerke mit über 1000 Slaves finden zum Beispiel in einem Flugzeug bei der Kabinenvernetzung, wie etwa der Kabinenbeleuchtung, In-Flight-Management, etc., Anwendung. In diesem Fall steigt die Rechenbelastung des Masters entsprechend der Anzahl der von ihm gesteuerten Slaves an.

[0006] Ein Master in einem solchen umfangreichen Master-Slave Computernetzwerk muss in dem ihm vorgegebenen Zeitraster die Steuerbefehle für sämtliche Slaves erzeugen und an die jeweiligen Slaves übertragen. Für das Erzeugen und Übermitteln der Steuerbefehle ist ein sehr hoher Rechenaufwand des Masters erforderlich. Je nach Priorität kann der Master in dieser Zeit keine weiteren Rechenaufgaben ausführen oder die Steuerung der Slaves erfolgt entsprechend zeitverzögert.

[0007] Der Rechenaufwand zum Erzeugen der Steuerbefehle skaliert dabei annähernd linear mit der Anzahl der von dem Master zu steuernden Slaves. Benötigt ein Master zum Erzeugen eines Steuerbefehls für einen Slave z.B. 5 ms, so benötigt der Master zum Erzeugen der Steuerbefehle für 100 Slaves bereits ca. 500 ms und bei 1000 Slaves sogar 5 s.

[0008] Verfügt der Master eines Netzwerks nicht über genügend Rechenressourcen, um neben dem Erzeugen der Steuerbefehle zusätzliche Berechnungen in einer vorgegebenen Zeit durchzuführen, führt dies zwangsläufig zu verzögerten Reaktionen des Masters, was bei der Steuerung der Teilnehmer eines Computernetzwerks in einem Flugzeug z.B. Komforteinbußen bewirkt.

[0009] Um parallel zum Erzeugen der Steuerbefehle für die Slaves dennoch in der Lage zu sein, weitere Rechenaufgaben durchzuführen, werden daher meist mehrere Recheneinrichtungen verwendet, die z.B. unabhängig von dem Master die notwendigen Berechnungen ausführen. Der Einsatz weiterer Recheneinrichtungen erhöht jedoch die Komplexität und vor allem auch die Kosten eines Computernetzwerks. Zudem steigen dadurch auch der Installationsaufwand und der Wartungsaufwand sowie die damit verbundenen Kosten.

[0010] Die US 2002 / 0 112 070 A1 und die US 2008 / 0 189 459 A1 zeigen Systeme, bei welchem ein Netzwerkkontroller Gruppen von Netzwerkteilneh-

mern ansprechen kann. Das Dokument „Efficient Polling of Devices in CANopen Networks“ offenbart ebenfalls, dass ein Bus-Master mehrere Bus-Teilnehmer in Gruppen ansprechen kann.

Zusammenfassung der Erfindung

[0011] Vor diesem Hintergrund ist es eine Aufgabe der vorliegenden Erfindung, bei einem Master-Slave Computernetzwerk die Master-Recheneinrichtung in ihrer Rechenbelastung zu entlasten.

[0012] Erfindungsgemäß wird diese Aufgabe durch eine Steuervorrichtung mit den Merkmalen des Patentanspruchs 1, ein Computernetzwerk mit den Merkmalen des Patentanspruchs 6 sowie ein Verfahren mit den Merkmalen des Patentanspruchs 8 gelöst.

[0013] Die der vorliegenden Erfindung zu Grunde liegende Erkenntnis besteht darin, dass das sequentielle Bereitstellen und Übermitteln der Programme für alle Netzwerkteilnehmer vor allem bei Vorhandensein einer großen Anzahl an Netzwerkteilnehmern ein sehr rechenintensiver Prozess ist. Die der vorliegenden Erfindung zu Grunde liegende Idee besteht nun darin, den Prozess des Erzeugens und Übermittels der Programme neu zu strukturieren und weitestgehend von der Master-Recheneinrichtung zu entkoppeln. Dabei wird eine Kontrolllogik bereitgestellt, die die Master-Recheneinrichtung unterstützen soll, indem sie insbesondere das Übermitteln der Programme an die Netzwerkteilnehmer übernimmt und die Master-Recheneinrichtung damit zumindest teilweise entlastet.

[0014] Erfindungsgemäß erzeugt die Master-Recheneinrichtung eine Reihe von Programmen, welche mehrere Programmbefehle enthalten können und welche an die Netzwerkteilnehmer übermittelt werden sollen. Ferner bildet die Master-Recheneinrichtung für jeden Netzwerkteilnehmer eine Zuordnung zwischen dem entsprechenden Netzwerkteilnehmer und einem der erzeugten Programme. Vorteilhafterweise erzeugt der Hauptrechner die Programme und Zuordnungen beim Hochfahren des Masters oder des Netzwerks.

[0015] Die Kontrolllogik ermittelt anhand der Zuordnungen, welches Programm welchem Netzwerkteilnehmer entspricht und gruppiert Netzwerkteilnehmer, welchen das gleiche Programm zugewiesen ist in gemeinsame Gruppen. Daraufhin übermittelt die Kontrolllogik die jeweiligen Programme an die entsprechenden Gruppen von Netzwerkteilnehmern. Da nunmehr die Kontrolllogik, und nicht die Master-Recheneinrichtung, das Übermitteln der Programme durchführt, steht die Recheneinrichtung, während die Kontrolllogik die Programme übermittelt, für weitere Rechenvorgänge zur Verfügung.

[0016] Die Master-Recheneinrichtung ist erfindungsgemäß lediglich dazu vorgesehen, neue Programme und Zuordnungen zu erzeugen und in dem Speicher abzulegen, falls die vorherigen Programme und Zuordnungen geändert werden sollen. Dies geschieht in der Regel lediglich einmal unmittelbar nach dem Hochfahren des Masters oder des Computernetzwerks. Die Master-Recheneinrichtung muss ein Programm, welches mehr als einem Netzwerkteilnehmer übermittelt werden soll, nur ein einziges Mal erstellen und dann den entsprechenden Netzwerkteilnehmern zuordnen. Dies entlastet die Master-Recheneinrichtung weiter. Nach dem Erzeugen der Programme und Zuordnungen werden die Rechenressourcen der Master-Recheneinrichtung nicht weiter durch das Übermitteln der Programme belastet und können für andere Aufgaben eingesetzt werden.

[0017] Die Kontrolllogik ist insbesondere dazu ausgebildet, Programme, welche mehr als einem Netzwerkteilnehmer zugeordnet werden sollen, parallel an diese Netzwerkteilnehmer zu übermitteln. Dabei kann die Kontrolllogik die Aufbereitung der Programme parallel ausführen und/oder das eigentliche Übermitteln der Programme über ein Netzwerk parallelisieren. Wird das Übermitteln der einzelnen Programme über ein Netzwerk parallel durchgeführt, so wird ein Programm nur ein einziges Mal übertragen und dabei an alle entsprechenden Netzwerkteilnehmer gleichzeitig adressiert. Die parallele Übermittlung der Programme entlastet dabei die Kontrolllogik, welche Programme nur ein einziges Mal aufbereiten muss, auch wenn diese an mehrere Netzwerkteilnehmer übermittelt werden müssen. Ferner wird auch das Netzwerk entlastet, da ein Programm, welches an mehrere Netzwerkteilnehmer übermittelt werden muss, lediglich an mehrere Netzwerkteilnehmer adressiert wird und nicht mehrmals physikalisch über das Netzwerk übertragen wird.

[0018] Vorteilhafte Ausführungsformen und Weiterbildungen ergeben sich aus den Unteransprüchen sowie aus der Beschreibung unter Bezugnahme auf die Figuren.

[0019] In einer Ausführungsform ist ein Speicher vorgesehen, in welchem die den Netzwerkteilnehmern zugeordneten Programme sowie eine Gruppeninformation der Zuordnung der Programme zu den jeweiligen Netzwerkteilnehmern und eine Längeninformaton über die jeweilige Programmlänge abgelegt sind. Wird ein eigener Speicher für diese Informationen bereit gestellt, kann die Kontrolllogik jederzeit und unabhängig von der Master-Recheneinrichtung auf beliebige Bestandteile der Programme, der Gruppeninformation, der Zuordnungen und der Längeninformaton zugreifen. Dadurch wird eine effiziente und unabhängige Übermittlung der

Programme durch die Kontrolllogik an die Netzwerkteilnehmer ermöglicht.

[0020] In einer weiteren Ausführungsform weist der Speicher einen Programm-Speicher auf, der dazu ausgebildet ist, die den Netzwerkteilnehmern zugeordneten Programme zu speichern. Der Speicher weist ferner einen Zuordnungs-Speicher auf, der dazu ausgebildet ist, die mindestens eine Zuordnung der Netzwerkteilnehmer zu einer Gruppe von Netzwerkteilnehmern und die Programmlängen der jeweiligen Programme zu speichern. Wird der Speicher in einen Programm-Speicher und einen Zuordnungs-Speicher untergliedert, so wird eine geordnete und strukturierte Ablage der Programme, der Gruppeninformation der Zuordnungen und der Längeninformation möglich. Dadurch können die Master-Rechen-einrichtung und die Kontrolllogik effizient auf diese Daten zugreifen.

[0021] In einer weiteren Ausführungsform sind der Programm-Speicher und der Zuordnungs-Speicher physikalisch in einem gemeinsamen Speicher angeordnet, beispielsweise in einem DRAM oder SRAM, und nur logisch voneinander getrennt, zum Beispiel als separate Speicherbereiche eines gemeinsamen Speicher-Adressraums, ausgebildet. In einer dazu alternativen Ausführungsform sind der Programm-Speicher und der Zuordnungs-Speicher physikalisch getrennt als separate Speicher ausgeführt.

[0022] Typischerweise, jedoch nicht notwendigerweise besteht ein einer Gruppe von Netzwerkteilnehmern zugewiesenes Programm aus einer Anzahl von Programmbefehlen, wobei die Kontrolllogik dazu ausgebildet ist, die Anzahl an Programmbefehlen entsprechend einer vorgegebenen Reihenfolge nacheinander an die Netzwerkteilnehmer zu übermitteln. Ein Programm in eine Vielzahl von Programmbefehlen zu unterteilen und diese Programmbefehle in einer vorgegebenen Reihenfolge an die Netzwerkteilnehmer zu übertragen, erlaubt eine flexible Gestaltung der einzelnen Programme und damit eine flexible Steuerung der Netzwerkteilnehmer.

[0023] In einer weiteren Ausführungsform ist jeweils ein Programmzähler vorgesehen, welcher fortwährend durch Zählen die Zahl der jeweils übermittelten Programmbefehle eines Programms ermittelt. Ferner ist in einer weiteren Ausführungsform die Kontrolllogik dazu ausgebildet, die Übermittlung des Programms an die Netzwerkteilnehmer zu stoppen, sofern die durch den Programmzähler ermittelte Zahl der übermittelten Programmbefehle dieses Programms der im Zuordnungsspeicher abgelegten Längeninformation der entsprechenden Programmlänge dieses Programms entspricht. Dies ermöglicht eine exakte Steuerung der Übermittlung der Programme an die Netzwerkteilnehmer durch die Kontrolllogik, d.h. es ist stets sichergestellt, dass jeder

Befehl bzw. jede Instruktion eines Programms auch an den Netzwerkteilnehmer übermittelt wurde.

[0024] In einer weiteren Ausführungsform weist ein Programmbefehl mindestens ein einzelnes interpretierbares Kommando und/oder mindestens eine einzelne ausführbare Instruktion auf. Werden einem Netzwerkteilnehmer interpretierbare Kommandos übermittelt, so interpretiert eine auf diesem Netzwerkteilnehmer vorgesehene Funktionalität diese Kommandos entsprechend und reagiert basierend auf dem jeweiligen Kommando mit einer entsprechenden Reaktion. Werden einem Netzwerkteilnehmer hingegen ausführbare Instruktionen übermittelt, so kann der Netzwerkteilnehmer diese speichern und basierend auf diesen Instruktionen eine neue Funktionalität annehmen. Dieser Vorgang wird auch als Flash-Programmierung, Flash-Update oder auch nur Update bezeichnet.

[0025] In einer weiteren Ausführungsform ist die Kontrolllogik dazu ausgebildet, Programme einmalig und/oder zyklisch zu übermitteln. Werden Programme einmalig an die Netzwerkteilnehmer übertragen, können so effektiv die Programme durchgeführt werden, die zum Beispiel zur Inbetriebnahme eines Netzwerkteilnehmers notwendig sind, während des Betriebs eines Netzwerkteilnehmers aber nicht benötigt werden. Eine zyklische Übermittlung von Programmen ermöglicht es dagegen, eine Funktion eines Netzwerkteilnehmers wiederholt auszuführen. Wird dafür jeweils das gleiche Programm mehrmals übermittelt, können Speicherressourcen für das wiederholte Ablegen des Programms eingespart werden.

[0026] In einer Ausführungsform sind mehr als 100, insbesondere mehr als 1000 Netzwerkteilnehmern vorgesehen, die über ein Bussystem mit der Steuervorrichtung verbunden sind. Die erfindungsgemäße Steuervorrichtung ist somit an Anforderungen moderner Computernetzwerke angepasst und ist sehr effektiv in der Lage, die einzelnen Netzwerkteilnehmer ohne allzu große Verzögerungen anzusprechen.

[0027] Die obigen Ausgestaltungen und Weiterbildungen lassen sich, sofern sinnvoll, beliebig miteinander kombinieren. Weitere mögliche Ausgestaltungen, Weiterbildungen und Implementierungen der Erfindung umfassen auch nicht explizit genannte Kombinationen von zuvor oder im Folgenden bezüglich der Ausführungsbeispiele beschriebenen Merkmale der Erfindung. Insbesondere wird dabei der Fachmann auch Einzelaspekte als Verbesserungen oder Ergänzungen zu der jeweiligen Grundform der vorliegenden Erfindung hinzufügen.

Inhaltsangabe der Zeichnungen

[0028] Die vorliegende Erfindung wird nachfolgend anhand der in den schematischen Figuren der Zeichnungen angegebenen Ausführungsbeispiele näher erläutert. Es zeigen dabei:

Fig. 1 ein Ausführungsbeispiel einer erfindungsgemäßen Steuervorrichtung;

Fig. 2 ein weiteres Ausführungsbeispiel einer erfindungsgemäßen Steuervorrichtung;

Fig. 3 ein Ausführungsbeispiel eines erfindungsgemäßen Computernetzwerks;

Fig. 4 ein Ablaufdiagramm eines erfindungsgemäßen Verfahrens.

[0029] Die beiliegenden Zeichnungen sollen ein weiteres Verständnis der Ausführungsformen der Erfindung vermitteln. Sie veranschaulichen Ausführungsformen und dienen im Zusammenhang mit der Beschreibung der Erklärung von Prinzipien und Konzepten der Erfindung. Andere Ausführungsformen und viele der genannten Vorteile ergeben sich im Hinblick auf die Zeichnungen. Die Elemente der Zeichnungen sind nicht notwendigerweise maßstabsgetreu zueinander gezeichnet.

[0030] In allen Figuren sind gleiche bzw. funktionsgleiche Elemente und Vorrichtungen - sofern nichts Anderes angegeben ist - mit denselben Bezugszeichen versehen worden.

Beschreibung von Ausführungsbeispielen

[0031] **Fig. 1** zeigt ein Ausführungsbeispiel einer erfindungsgemäßen Steuervorrichtung 1. Die Steuervorrichtung 1 weist eine Master-Recheneinrichtung 2 auf, welche Programme $P_1 - P_m$ für Netzwerkteilnehmer $T_1 - T_n$ bereitstellt. Ferner weist die Steuervorrichtung 1 eine Kontrolllogik 3 auf, welche mit der Master-Recheneinrichtung 2 gekoppelt ist, und dazu ausgebildet ist, die Programme $P_1 - P_m$ an die Netzwerkteilnehmer $T_1 - T_n$ zu übermitteln.

[0032] Die Master-Recheneinrichtung 2 ist als eine programmgesteuerte Einrichtung ausgebildet, beispielsweise als Mikrocontroller, Mikroprozessor, digitaler Signalprozessor oder dergleichen, jedoch könnte dieser auch in Form einer programmierbaren Logikschaltung, wie etwa einem PLD oder FPGA, ausgebildet sein. In einer weiteren Ausführungsform kann die Master-Recheneinrichtung 2 auch als Computer-Programm-Produkt-Komponente eines Computers ausgeführt sein. Die Master-Recheneinrichtung 2 stellt jeweils mindestens ein Programm $P_1 - P_m$ für jeweils einen Netzwerkteilnehmer $T_1 - T_n$ bereit. Die Master-Recheneinrichtung 2 kann auch dasselbe Programm $P_1 - P_m$ für mehr als einen Netzwerkteilnehmer $T_1 - T_n$ bereitstellen.

[0033] In der in **Fig. 1** dargestellten Ausführungsform ist die Kontrolllogik 3 als ein anwendungsspezifischer Schaltkreis (ASIC) ausgebildet, welcher die Programme $P_1 - P_m$ von der Master-Recheneinrichtung 2 erhält. Die Kontrolllogik 3 kann in einer weiteren Ausführungsform auch als ein PLD, CPLD, FPGA, DSP oder dergleichen ausgebildet sein. Die Kontrolllogik 3 gruppiert die Netzwerkteilnehmer $T_1 - T_n$ gemäß den bereitgestellten Programmen $P_1 - P_m$ und übermittelt den Netzwerkteilnehmern $T_1 - T_n$ die einzelnen Programme $P_1 - P_m$. Sind zwei oder mehr Netzwerkteilnehmer $T_1 - T_n$ in einer gemeinsamen Gruppe eingruppiert, so übermittelt die Kontrolllogik 3 den gemeinsam gruppierten Netzwerkteilnehmern $T_1 - T_n$ ein jeweiliges Programm $P_1 - P_m$ parallel.

[0034] Die Verbindung zwischen der Master-Recheneinrichtung 2 und der Kontrolllogik 3 ist in **Fig. 1** als eine direkte parallele digitale Verbindung ausgeführt, über welche die Master-Recheneinrichtung 2 der Kontrolllogik 3 die Programme $P_1 - P_m$ bereitstellt. In einer weiteren Ausführungsform kann die Verbindung auch als eine serielle digitale Verbindung ausgeführt werden.

[0035] **Fig. 2** zeigt ein weiteres Ausführungsbeispiel einer erfindungsgemäßen Steuervorrichtung 1. **Fig. 2** unterscheidet sich von **Fig. 1** dahingehend, dass die Verbindung zwischen der Master-Recheneinrichtung 2 und der Kontrolllogik 3 keine direkte Punkt-zu-Punkt Verbindung darstellt. Vielmehr ist ein Speicher 4 vorgesehen, welcher zwischen der Master-Recheneinrichtung 2 und der Kontrolllogik 3 angeordnet ist und die Recheneinrichtung 2 und die Kontrolllogik 3 miteinander koppelt. Der Speicher 4 weist einen Programm-Speicher 5 und einen Zuordnungs-Speicher 6 auf. Zusätzlich ist die Kontrolllogik 3 in **Fig. 2** mit mindestens einem Programmzähler 7 gekoppelt. Der Programmzähler 7 ist hier mit einer durchgezogenen Linie dargestellt. Weitere, theoretisch mögliche Programmzähler 7 sind mit einer gestrichelten Linie dargestellt. Der Programmzähler 7 kann auch Bestandteil der Kontrolllogik 3 sein.

[0036] Der Speicher 4 in **Fig. 2** weist zwei separate Speicher, einen Programm-Speicher 5 und einen Zuordnungs-Speicher 6 auf. In der hier dargestellten Ausführungsform ist der Speicher 4 als paralleler RAM-Speicher (Random-Access-Memory) ausgeführt, auf den sowohl die Master-Recheneinrichtung 2 als auch die Kontrolllogik 3 zugreifen können. In der dargestellten Ausführungsform besteht der Speicher 4 aus zwei separaten Speicher-Bausteinen. In einer alternativen Ausführungsform sind der Programm-Speicher 5 und der Zuordnungs-Speicher 6 in einem gemeinsamen Speicherbaustein vorgesehen. In weiteren Ausführungsformen kann der Speicher 4 als zumindest ein serieller Speicherbaustein, zum Beispiel ein I²C- oder ein SPI-Speicherbaustein, vorgesehen sein. Denkbar wäre auch, dass der Spei-

cher 4 als ein Festplattenspeicher, ein Speicherlaufwerk, zum Beispiel ein CD-Laufwerk mit Schreibfunktion, oder dergleichen ausgeführt ist. In diesen Fällen sind der Programm-Speicher 5 und der Zuordnungsspeicher 6 über eine IDE-Schnittstelle, eine SATA-Schnittstelle, eine eSATA-Schnittstelle, eine Firewire-Schnittstelle, eine USB-Schnittstelle oder dergleichen mit dem Speicher 4 gekoppelt.

[0037] Der Programm-Speicher 5 des Speichers 4 in **Fig. 2** ist dazu ausgebildet, die von der Master-Recheneinrichtung 2 bereitgestellten Programme $P_1 - P_m$ zu speichern. Die Programme $P_1 - P_m$ bestehen aus einzelnen Programmbefehlen $Pb_1 - Pb_x$. Dabei kann ein einzelnes Programm $P_1 - P_m$ zum Beispiel aus fünf Programmbefehlen $Pb_1 - Pb_5$ bestehen und ein zweites Programm $P_1 - P_m$ aus 10 Programmbefehlen $Pb_5 - Pb_{15}$ bestehen. In einer weiteren Ausführungsform können sich Programme $P_1 - P_m$ in dem Speicher 4 auch überschneiden. Ein erstes Programm $P_1 - P_m$ kann beispielsweise aus 15 Programmbefehlen $Pb_1 - Pb_{15}$ bestehen. Ein zweites Programm $P_1 - P_m$ kann dabei ein Teilprogramm $P_1 - P_m$ des ersten Programms $P_1 - P_m$ bilden, welches aus den Programmbefehlen Pb_5 bis Pb_{10} besteht.

[0038] Der mindestens eine Programmzähler 7 in **Fig. 2** ist als externer Zähler der Kontrolllogik 3 dargestellt. In einer weiteren Ausführungsform ist der Programmzähler 7 als ein internes Element der Kontrolllogik 3 oder als ein Bestandteil des Speichers 4 ausgeführt. In einer Ausführungsform ist ein Programmzähler 7 pro Gruppe von Netzwerkteilnehmern $T_1 - T_n$ vorgesehen.

[0039] **Fig. 3** zeigt ein Ausführungsbeispiel eines erfindungsgemäßen Computernetzwerks 10. Das Computernetzwerk 10 weist eine erfindungsgemäße Steuervorrichtung 1 auf, wie sie zum Beispiel in den **Fig. 1** und **Fig. 2** dargestellt ist. Die Steuervorrichtung 1 ist hier über einen Netzwerkbus 11 mit den Netzwerkteilnehmern $T_1 - T_n$ gekoppelt. Die Netzwerkteilnehmer T_1, T_2, T_3 und T_4 sind direkt mit dem Netzwerkbus 11 gekoppelt. Die Netzwerkteilnehmer T_5, T_6 und T_n sind mit dem Netzwerkteilnehmer T_4 gekoppelt, der dadurch die Funktion eines Stern-Verteilers, auch Hub oder Switch genannt, einnimmt.

[0040] In einem Ausführungsbeispiel sind in dem Computernetzwerk 10 über 100 Netzwerkteilnehmer $T_1 - T_n$ vorgesehen. In einem weiteren Ausführungsbeispiel sind in dem Computernetzwerk 1000 oder mehr Netzwerkteilnehmer $T_1 - T_n$ vorgesehen. In noch einem weiteren Ausführungsbeispiel sind in dem Computernetzwerk mehr als eine Steuervorrichtung 1 vorgesehen. Beispielsweise kann eine Steuervorrichtung 1 eine erste Menge von 1000 Netzwerkteilnehmern $T_1 - T_n$ steuern und eine zweite

Steuervorrichtung 1 eine zweite Menge von 1000 Netzwerkteilnehmern $T_1 - T_n$ steuern.

[0041] Der Netzwerkbus 11 ist in der in **Fig. 3** dargestellten Ausführungsform als ein Ethernet-Netzwerkbus ausgebildet. In weiteren Ausführungsformen ist der Netzwerkbus 11 als ein CAN-Bus, ein ARINC-Bus, ein Profibus oder eine andere Feldbus-Architektur ausgebildet. Der Netzwerkbus 11 kann als elektrischer, optischer oder kabelloser bzw. Funk-, zum Beispiel als ZigBee-, Netzwerkbus, etc., ausgebildet sein.

[0042] **Fig. 4** zeigt ein Ablaufdiagramm eines erfindungsgemäßen Verfahrens.

[0043] In einem ersten Schritt S1 werden eine Master-Recheneinrichtung 2 und eine Vielzahl von Netzwerkteilnehmern $T_1 - T_n$ eines Computernetzwerks 10 bereitgestellt.

[0044] In einem zweiten Schritt S2 werden die Netzwerkteilnehmer $T_1 - T_n$, welche von der Master-Recheneinrichtung 2 mit denselben Programmen $P_1 - P_m$ angesprochen werden sollen, zu jeweils einer Gruppe an Netzwerkteilnehmern $T_1 - T_n$ gruppiert. Die Kontrolllogik 3 gruppiert die Netzwerkteilnehmer $T_1 - T_n$ gemäß den bereitgestellten Programmen $P_1 - P_m$. Ist ein einzelnes Programm $P_1 - P_m$ zwei oder mehreren Netzwerkteilnehmern $T_1 - T_n$ zugewiesen, so werden diese Netzwerkteilnehmer $T_1 - T_n$ von der Kontrolllogik 3 in eine gemeinsame Gruppe gruppiert.

[0045] In einem letzten Schritt S3 des erfindungsgemäßen Verfahrens wird ein Programm $P_1 - P_m$, welches von der Master-Recheneinrichtung 2 jeweils an dieselben Netzwerkteilnehmer $T_1 - T_n$ übermittelt werden soll, an die Netzwerkteilnehmer $T_1 - T_n$ derselben Gruppe parallel übermittelt. Parallel bedeutet in diesem Zusammenhang, dass die Kontrolllogik die Vorbereitungen und Berechnungen zum Übermitteln eines Programms $P_1 - P_m$ für jede Gruppe nur einmal durchführt. Die Übermittlung der Programme $P_1 - P_m$ findet dann abhängig von dem gewählten Netzwerktyp, welcher die Netzwerkteilnehmer $T_1 - T_n$ mit der Steuervorrichtung 1 verbindet parallel, seriell oder in einer einzigen Botschaft statt. Findet die Übermittlung über das Netzwerk 10 parallel statt, werden mehrere Botschaften mit gleichem Inhalt an unterschiedliche Empfänger gleichzeitig übermittelt. Findet die Übermittlung seriell statt, werden die Botschaften für die einzelnen Netzwerkteilnehmer $T_1 - T_n$ nacheinander auf dem Netzwerk 10 übertragen. Findet die Übermittlung in einer einzigen Botschaft statt, so wird die eine Botschaft an mehrere Netzwerkteilnehmer adressiert und einmalig übermittelt, wie zum Beispiel in einer IP-Multicast Übermittlung.

[0046] Obwohl die vorliegende Erfindung anhand bevorzugter Ausführungsbeispiele vorstehend beschrieben wurde, ist sie darauf nicht beschränkt, sondern auf vielfältige Art und Weise modifizierbar. Insbesondere lässt sich die Erfindung in mannigfaltiger Weise verändern oder modifizieren, ohne vom Kern der Erfindung abzuweichen.

[0047] Die vorliegende Erfindung sei nicht auf die Anwendung im Kabinenbereich eines Passagierflugzeugs beschränkt, sondern lässt sich z.B. auch sehr vorteilhaft in der Automatisierungstechnik, z.B. der Gebäudeautomatisierung oder der Fahrzeugautomatisierung, bei denen häufig ebenfalls mehr als 1000 Slaves pro Master vorhanden sind, verwenden.

Bezugszeichenliste

| | |
|-------------|--------------------------|
| 1 | Steuervorrichtung |
| 2 | Master-Recheneinrichtung |
| 3 | Kontrolllogik |
| 4 | Speicher |
| 5 | Programm-Speicher |
| 6 | Zuordnungs-Speicher |
| 7 | Programmzähler |
| 10 | Computernetzwerk |
| 11 | Netzwerkbus |
| $P_1 - P_m$ | Programm |
| Pb_1 | Programmbefehl |
| Pb_x | Programmbefehl |
| S1-S3 | Verfahrensschritt |
| $T_1 - T_n$ | Netzwerkteilnehmer |

Patentansprüche

1. Steuervorrichtung (1) zum Steuern einer Vielzahl von Netzwerkteilnehmern ($T_1 - T_n$) eines Computernetzwerks (10), mit einer Master-Recheneinrichtung (2) zum Steuern der Netzwerkteilnehmer ($T_1 - T_n$) entsprechend einem Programm, welche jedem Netzwerkteilnehmer ($T_1 - T_n$) mindestens ein Programm ($P_1 - P_m$) zuweist, mit einer Kontrolllogik (3), welche dazu ausgelegt ist, die Netzwerkteilnehmer ($T_1 - T_n$) entsprechend den diesen Netzwerkteilnehmern ($T_1 - T_n$) jeweils zugewiesenen Programmen ($P_1 - P_m$) zu gruppieren und denjenigen Netzwerkteilnehmern ($T_1 - T_n$) derselben Gruppe, denen von der Master-Recheneinrichtung (2) dasselbe Programm ($P_1 - P_m$) zugewiesen ist, dieses Programm ($P_1 - P_m$) parallel zu übermitteln, wobei ein Speicher (4) vorgesehen ist, in welchem die den Netzwerkteilnehmern ($T_1 - T_n$) zugeordneten

Programme ($P_1 - P_m$), eine Gruppeninformation der Zuordnung der Programme ($P_1 - P_m$) zu den jeweiligen Netzwerkteilnehmern ($T_1 - T_n$) und eine Längeninformation über die jeweilige Programmlänge abgelegt sind; und

wobei der Speicher (4) einen Programm-Speicher (5) aufweist, der dazu ausgebildet ist, die den Netzwerkteilnehmern ($T_1 - T_n$) zugewiesenen Programme $P_1 - P_m$ zu speichern, und dass der Speicher (4) einen Zuordnungs-Speicher (6) aufweist, der dazu ausgebildet ist, die mindestens eine Zuordnung der Netzwerkteilnehmer ($T_1 - T_n$) zu einer Gruppe von Netzwerkteilnehmern ($T_1 - T_n$) und die Programmlängen der diesen jeweils zugewiesenen Programme ($P_1 - P_m$) zu speichern; wobei die Master-Recheneinrichtung (2) ausgebildet ist, neue Programme ($P_1 - P_m$) und Zuordnung der Programme ($P_1 - P_m$) zu erzeugen und in dem Speicher (4) abzulegen, falls die vorherigen Programme ($P_1 - P_m$) und Zuordnung der Programme ($P_1 - P_m$) geändert werden sollen.

2. Steuervorrichtung nach einem der vorherigen Ansprüche, **dadurch gekennzeichnet**, dass ein einer Gruppe von Netzwerkteilnehmern ($T_1 - T_n$) zugewiesenes Programm ($P_1 - P_m$) aus einer Anzahl von Programmbefehlen ($Pb_1 - Pb_x$) besteht, wobei die Kontrolllogik (3) dazu ausgebildet ist, die Anzahl an Programmbefehlen ($Pb_1 - Pb_x$) entsprechend einer vorgegebenen Reihenfolge nacheinander an die Netzwerkteilnehmer ($T_1 - T_n$) zu übermitteln.

3. Steuervorrichtung nach Anspruch 2, **dadurch gekennzeichnet**, dass jeweils ein Programmzähler (7) vorgesehen ist, welcher fortwährend die Zahl der übermittelten Programmbefehle ($Pb_1 - Pb_x$) eines Programms ($P_1 - P_m$) zählt, und dass die Kontrolllogik (3) dazu ausgebildet ist, die Übermittlung des Programms ($P_1 - P_m$) an die Netzwerkteilnehmer ($T_1 - T_n$) zu stoppen, sofern die durch den Programmzähler (7) ermittelte Zahl der übermittelten Programmbefehle ($Pb_1 - Pb_x$) dieses Programms ($P_1 - P_m$) der im Zuordnungsspeicher abgelegten Längeninformation der entsprechenden Programmlänge dieses Programms ($P_1 - P_m$) entspricht.

4. Steuervorrichtung nach einem der Ansprüche 2 oder 3, **dadurch gekennzeichnet**, dass ein Programmbefehl ($Pb_1 - Pb_x$) mindestens ein einzelnes interpretierbares Kommando und/oder mindestens eine einzelne ausführbare Instruktion aufweist.

5. Steuervorrichtung nach einem der vorherigen Ansprüche, **dadurch gekennzeichnet**, dass die Kontrolllogik (3) dazu ausgebildet ist, Programme ($P_1 - P_m$) einmalig und/oder zyklisch zu übermitteln.

6. Computernetzwerk, insbesondere Computernetzwerk (10) in einem Flugzeug, welches als Master-Slave-Computernetzwerk (10) ausgebildet ist,

mit einer Vielzahl von Netzwerkteilnehmern ($T_1 - T_n$),
mit einer Steuervorrichtung (1) nach einem der vorherigen Ansprüche zum Steuern der Vielzahl von Netzwerkteilnehmern ($T_1 - T_n$).

7. Computernetzwerk (10) nach Anspruch 6, **dadurch gekennzeichnet**, dass mehr als 100, insbesondere mehr als 1000 Netzwerkteilnehmer ($T_1 - T_n$) vorgesehen sind, die über ein Bussystem mit der Steuervorrichtung verbunden sind.

8. Verfahren zum Betreiben eines Computernetzwerks (10), mit den Schritten:
Bereitstellen (S1) einer Master-Recheneinrichtung (2) und einer Vielzahl von Netzwerkteilnehmer ($T_1 - T_n$) eines Computernetzwerks (10);
Gruppieren (S2) der Netzwerkteilnehmer ($T_1 - T_n$), welche von der Master-Recheneinrichtung (2) mit demselben Programm ($P_1 - P_m$) angesprochen werden sollen, zu jeweils einer Gruppe an Netzwerkteilnehmern ($T_1 - T_n$);
Speichern der den Netzwerkteilnehmern ($T_1 - T_n$) zugewiesenen Programme ($P_1 - P_m$) in einen Programm-Speicher (5);
Speichern der Gruppeninformation der Zuordnung der Programme ($P_1 - P_m$) zu den jeweiligen Netzwerkteilnehmern ($T_1 - T_n$) und einer Längeninformatio n über die jeweilige Programmlänge in einen Zuordnungs-Speicher (6);
Paralleles Übermitteln (S3) eines Programms ($P_1 - P_m$), welches von der Master-Recheneinrichtung (2) jeweils an dieselben Netzwerkteilnehmer ($T_1 - T_n$) übermittelt werden soll, an die Netzwerkteilnehmern ($T_1 - T_n$) derselben Gruppe;
Erzeugen und Speichern neuer Programme ($P_1 - P_m$) und Zuordnung der Programme ($P_1 - P_m$), falls die vorherigen Programme ($P_1 - P_m$) und Zuordnung der Programme ($P_1 - P_m$) geändert werden sollen.

9. Verfahren nach Anspruch 8, **dadurch gekennzeichnet**, dass ein einer Gruppe von Netzwerkteilnehmern ($T_1 - T_n$) zugewiesenes Programm ($P_1 - P_m$) aus einer Anzahl von Programmbefehlen ($Pb_1 - Pb_x$) besteht und dass der Schritt des parallelen Übermittels eines Programms ($P_1 - P_m$) ferner aufweist, dass die Anzahl an Programmbefehlen ($Pb_1 - Pb_x$) entsprechend einer vorgegebenen Reihenfolge nacheinander an die Netzwerkteilnehmer ($T_1 - T_n$) übermittelt wird.

10. Verfahren nach einem der Ansprüche 8 bis 9, **dadurch gekennzeichnet**, dass die Zahl der übermittelten Programmbefehle ($Pb_1 - Pb_x$) eines Programms ($P_1 - P_m$) gezählt wird, wobei die Übermittlung des Programms ($P_1 - P_m$) an die Netzwerkteilnehmer ($T_1 - T_n$) gestoppt wird, sofern die durch Zählen ermittelte Zahl der übermittelten Programmbefehle ($Pb_1 - Pb_x$) dieses Programms ($P_1 - P_m$) der gespeicherten Längeninformatio n der entsprechen-

den Programmlänge dieses Programms ($P_1 - P_m$) entspricht.

11. Verfahren nach einem der Ansprüche 8 bis 10, **dadurch gekennzeichnet**, dass ein Programmbefehl ($Pb_1 - Pb_x$) mindestens ein einzelnes interpretierbares Kommando und/oder mindestens eine einzelne ausführbare Instruktion aufweist.

12. Verfahren nach einem der Ansprüche 8 bis 11, **dadurch gekennzeichnet**, dass Programme ($P_1 - P_m$) einmalig und/oder zyklisch übermittelt werden.

Es folgen 2 Seiten Zeichnungen

Anhängende Zeichnungen

Fig. 1

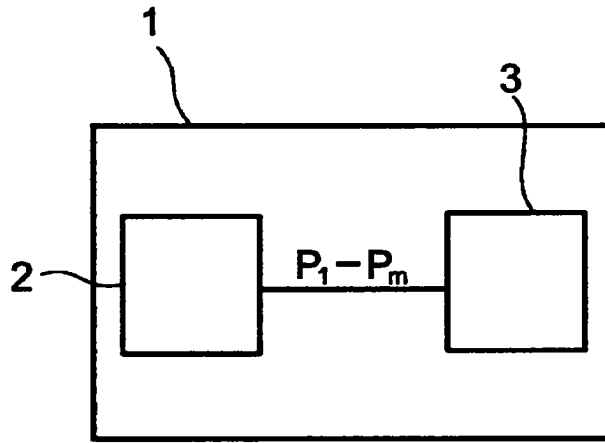


Fig. 2

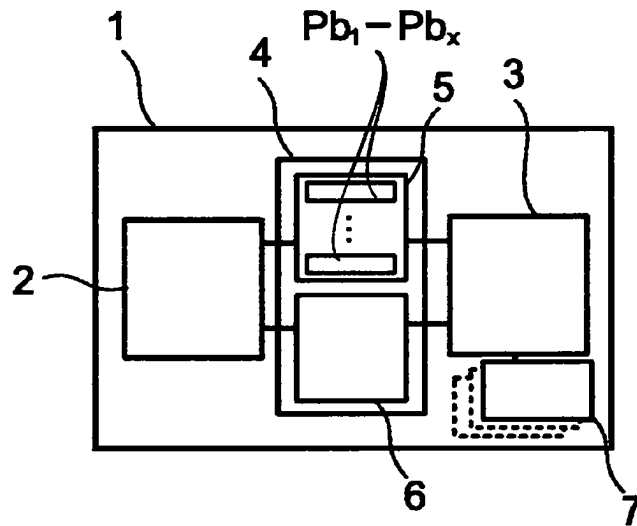


Fig. 3

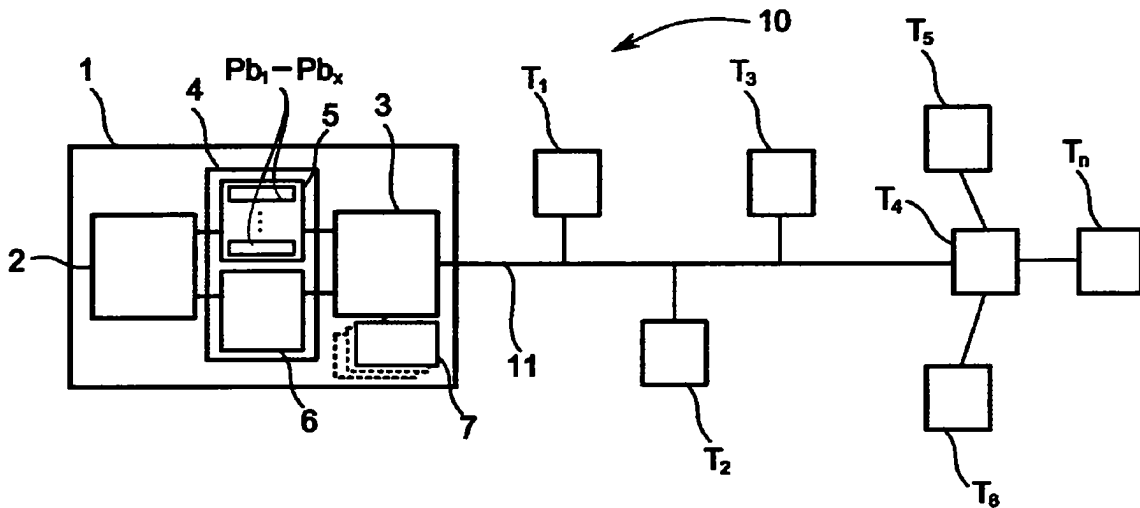


Fig. 4

