

(12)

# PATENTSCHRIFT

(21) Anmeldenummer: 1644/97  
(22) Anmeldetag: 26.09.1997  
(42) Beginn der Patentdauer: 15.06.2000  
(45) Ausgabetag: 26.02.2001

(51) Int. Cl.<sup>7</sup>: **G06F 13/42**

(56) Entgegenhaltungen:  
DE 4408488A1 EP 0658257A2 US 4866606A

(73) Patentinhaber:  
FTS COMPUTERTECHNIK GES.M.B.H.  
A-2500 BADEN-SIEGENFELD,  
NIEDERÖSTERREICH (AT).

## (54) METHODE UND RECHNERKNOTEN ZUR ERZIELUNG VON ZEITLICHER DATENKONSISTENZ IN VERTEILTEN ECHTZEITSYSTEMEN

**AT 407 308 B**

(57) Es ist das wesentliche Ziel der vorliegenden Erfindung die Genauigkeit der in einem verteilten Echtzeitcomputersystem erfaßten zeitabhängigen Meßwerte zu verbessern und damit die Qualität von regelungstechnischen Verfahren zu erhöhen.

Dieses Ziel wird dadurch erreicht, daß die dezentral erfaßten Meßdaten beim Sender derart aktualisiert werden, daß die Meßdaten zum Zeitpunkt der Verarbeitung beim Empfänger ein zeitrichtiges Abbild der durch die Daten zu vermittelnden sich nach bekannten Regeln ändernden Wirklichkeit wiedergeben und eine gewünschte Fehlertoleranz im Sensorsystem realisiert wird, ohne die Komplexität der Applikation zu erhöhen.

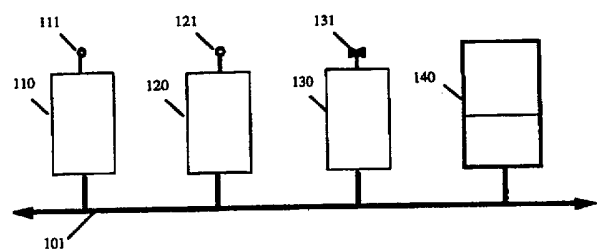


FIG. 1

## LITERATUR

- Zitierte Patente:
- Europäisches Patent:  
 0 658 257            18.12.96            Kopetz, H
- 5    Deutsches Patent:  
 DE 4408 488        14.3.1994            Eitrich, F.T.
- US Patent:  
 4,866,606            12.9.1989            Kopetz, H.
- 10    Internationale Patentanmeldung:  
 PCT/AT 93/00138    2.9.1993            Kopetz, H.
- Andere Veröffentlichungen:  
 Kopetz, H. (1995). TTP/A -- A Time-Triggered Protocol of Body Electronics Using Standard  
 UARTS. *Proc. SAE World Congress, Society of Automotive Engineers, SAE Technical Paper*  
 950039. pp. 1-9.
- 15    Kopetz, H. (1997). *Real-Time Systems, Design Principles for Distributed Embedded*  
*Applications*; ISBN: 0-7923-9894-7. Boston. Kluwer Academic Publishers.

## TECHNISCHES UMFELD

20    Diese Erfindung betrifft eine Methode und Rechnerknoten zur Erzielung der zeitlichen Konsis-  
 tenz von zeitabhängigen Echtzeitdaten in einem verteilten Echtzeitcomputersystem. Ein solches  
 verteiltes Echtzeitcomputersystem besteht aus einer Vielzahl von Sensorknoten, Verarbeitungsk-  
 noten, und Ausgabeknoten.

## HINTERGRUND DIESER ERFINDUNG

25    Die Entwicklung von hochintegrierten Systemchips (System on a Chip--SOC) führt dazu, daß  
 es wirtschaftlich sinnvoll wird, in einem Echtzeitsystem an jedem Sensor und jedem Aktuator einen  
 eigenen Knotenrechner zu installieren und die Verbindung dieser Knotenrechner mit einem  
 seriellen Bus zu realisieren. In regelungstechnischen Anwendungen ergibt sich Notwendigkeit, die  
 zeitabhängigen Sensor- und Aktuatordaten in einem verteilten Computersystem so zu übertragen,  
 30    daß der Jitter klein ist und die temporale und räumliche Konsistenz der Daten gegeben ist. In der  
 Literatur (siehe Kopetz, H. (1997). *Real-Time Systems, Design Principles for Distributed*  
*Embedded Applications*; ISBN: 0-7923-9894-7. Boston. Kluwer Academic Publishers, pp. 159-164)  
 sind viele solcher seriellen Bussysteme beschrieben. Die meisten dieser bekannt gewordenen  
 Bussysteme sind ereignisgesteuert und ermöglichen es im allgemeinen nicht, das Zeitintervall  
 35    zwischen dem Lesen eines Sensorwertes beim Sensorknoten und dem Empfang des Sensor-  
 wertes beim Verarbeitungsknoten *a priori* zu bestimmen.

In der Offenlegungsschrift DE 4408488 wird vorgeschlagen, dieses Problem in einem ereignis-  
 gesteuerten System dadurch zu lösen, daß bei Absendung einer Nachricht nicht diejenigen Daten,  
 die zum Zeitpunkt der Stellung des Sendeauftrags übergeben wurden, verwendet werden, sondern  
 40    statt dessen die von der Rechen/Steuereinheit seit dem Zeitpunkt der Abgabe des Sendeauftrages  
 inzwischen aktualisierten Daten. Da jedoch der Zeitpunkt des Absendens einer Nachricht nicht *a*  
*priori* bekannt ist, kann der Fall eintreten, daß das Absenden einer Nachricht unmittelbar vor der  
 Aktualisierung der Daten erfolgt. Die aktualisierten Daten können dann nicht mehr berücksichtigt  
 werden.

45    Für die Übertragung von periodischen regelungstechnischen Echtzeitdaten sind zeitgesteuerte  
 Kommunikationsprotokolle von großem Vorteil, da diese Protokolle eine globale Zeitbasis von  
 hoher Genauigkeit aufbauen, die Zeitpunkte der Übertragung *a priori* bekannt sind und der Jitter  
 der Übertragung gering ist. Weiters kann in zeitgesteuerten Protokollen der Nachrichtenname aus  
 dem Zeitpunkt der Übertragung abgeleitet werden, so daß der Nachrichteninhalt sehr kurz ist, was  
 50    zu einer hohen Dateneffizienz der Kommunikation führt (siehe Europäisches Patent: 0 658 257  
 vom 18.12.96).

Auf einem seriellen Bussystem müssen die Sensorwerte von unterschiedlichen Sensorknoten  
 zu verschiedenen Zeitpunkten gesendet werden. In vielen regelungstechnische Anwendungen  
 haben die Sensorwerte eine unterschiedliche Dynamik, d.h., einige Sensorwerte (z.B., Druck oder  
 55    Lage) können sich sehr schnell ändern, andere wieder (z.B., Temperatur) ändern sich relativ lang-

sam. Es ist deshalb sinnvoll über ein serielles Bussystem die weniger dynamischen Sensorwerte zuerst zu übertragen und die Sensorwerte mit hoher Dynamik erst unmittelbar vor der Bearbeitung zu erfassen und an den Verarbeitungsknoten zu übertragen.

5 Wenn ein regelungstechnischer Algorithmus, der in einem Knotenrechner exekutiert wird, mehrere Sensordaten benötigt die von dezentralen Sensorknoten räumlich verteilt erfaßt werden, so sollen sich alle Sensordaten auf den *gleichen* Zeitpunkt beziehen, um die temporale Konsistenz der Sensorwerte sicher zu stellen.

Die beschriebenen, an sich widersprüchlichen Anforderungen werden in der vorliegenden Erfindung dadurch gelöst, daß

- 10 (i) das aus dem in zeitgesteuerten Protokollen *a priori* bekannten Zeitintervall zwischen dem Lesen eines Analogwertes am Sensorknoten und der Ankunft des Analogwertes am Verarbeitungsknoten eine dynamische Berechnung des zukünftigen Ankunftszeitpunktes am Verarbeitungsknoten durch den Sensorknoten durchgeführt wird,
- 15 (ii) die Sensorknoten die erfaßten Meßwerte derart aktualisieren, daß der Inhalt der Nachricht zum Zeitpunkt der Ankunft der Nachricht beim Empfänger ein weitgehend zeitrichtiges Abbild der durch die Nachricht zu vermittelnden sich nach bekannten Regeln ändernden Wirklichkeit wiedergibt.
- (iii) die Sensordaten kompakt in Ein-Byte oder Zwei-Byte Nachrichten kodiert werden, um den Zeitaufwand für die Übertragung zu minimieren.
- 20 (iv) unterschiedliche Beobachtungen desselben Tatbestandes durch mehrere Sensorknoten bereits im Kommunikationssystem abgeglichen werden, um Fehler zu erkennen und dem Bearbeitungsrechner ein einheitliches zeitrichtiges Bild der Wirklichkeit anzubieten, sodaß ein fehlertolerantes Sensorsystem ohne Zunahme der Anwendungs-komplexität realisiert werden kann.

25

#### ZUSAMMENFASSUNG

Es ist das wesentliche Ziel der vorliegenden Erfindung die zeitliche Konsistenz und damit die Genauigkeit der in einem verteilten Echtzeitcomputersystem erfaßten zeitabhängigen Analogmeßwerte zu verbessern und damit die Qualität von regelungstechnischen Verfahren zu erhöhen.

30

Dieses Ziel wird dadurch erreicht, daß die dezentral erfaßten Meßdaten beim Sender derart aktualisiert werden, daß die Meßdaten zum Zeitpunkt der Verarbeitung beim Empfänger ein zeitrichtiges Abbild der durch die Daten zu vermittelnden sich nach bekannten Regeln ändernden Wirklichkeit wiedergeben und eine gewünschte Fehlertoleranz im Sensorsystem realisiert wird, ohne die Komplexität der Applikation zu erhöhen.

35

#### KURZE BESCHREIBUNG DER ABBILDUNGEN

Das vorab beschriebene Ziel und andere neue Eigenschaften der vorliegenden Erfindung werden in den angeführten Abbildungen erläutert.

40

Fig. 1 zeigt die Struktur eines verteilten Echtzeitcomputersystems mit vier Knotenrechnern, die über einen seriellen Kommunikationskanal verbunden sind.

Fig. 2 zeigt die Struktur eines Masterknoten, bestehend aus einer autonomen Kommunikationskontrolleinheit und einem Host Computer, die über das gemeinsame Speicherschnittstelle kommunizieren.

45

Fig. 3 zeigt die Struktur eines Sensorknoten, bestehend aus einer einfachen Kommunikationskontrolleinheit, einem Host Computer und einem Prozeßinterface.

Fig. 4 zeigt die Zeitintervalle zwischen dem Lesen von Sensorwerten, der Übertragung auf dem seriellen Bus, und dem Beginn der Verarbeitung der Sensorwerte beim Empfänger.

Fig. 5 zeigt das Datenformat für Analogmeßwerte.

50

Fig. 6 zeigt einen fehlertoleranten Anschluß eines Sensorsystems an replizierte Knoten eines verteilten fehlertoleranten Echtzeitsystems.

Fig. 7 zeigt den Aufbau einer Initialisierungsrunde.

#### BESCHREIBUNG EINER REALISIERUNG

55

Im folgenden Abschnitt wird eine Realisierung des neuen Verfahrens an einem Beispiel mit vier Knotenrechnern gezeigt. Die Objekte in den Abbildungen sind so numeriert, daß die erste der

dreistelligen Objektziffern immer die Bildnummer angibt.

Fig. 1 zeigt ein Echtzeitsystem mit einem seriellen Bus **101** und drei Sensorknoten **110**, **120**, und **130** und einen Masterknoten **140**. Jeder Knotenrechner verfügt innerhalb des gegebenen Ensembles über einen eindeutigen physikalischen Namen über den der Knotenrechner angesprochen werden kann. Dieser eindeutige physikalische Name, z.B., ein Zahl aus einem 8, 16, 32, 48 der 64 Bit großem Namensraum, wird dem Knotenrechner während der Produktion zugewiesen oder kann mittels eines Schalters nachträglich eingestellt werden. Alle Knotenrechner kommunizieren über den seriellen zeitgesteuerten Bus **101** miteinander. Sensorknoten **110** und **120** lesen die angeschlossenen Sensoren **111** und **121**. Sensorknoten **130** kontrolliert den angeschlossenen Aktor **131**. Der Masterknoten **140** berechnet den Regelalgorithmus. Er übernimmt gleichzeitig die Bus Master Funktion, falls eine solche vorgesehen ist.

Fig. 2 zeigt die innere Struktur des Masterknoten **140**. Er besteht aus zwei Subsystemen, dem intelligenten Kommunikationskontroller **220** und den Host Computer **230**. Falls erforderlich, kann noch ein zweiter Kommunikationskontroller vorhanden sein (siehe **640** bzw. **650**). Der intelligente Kommunikationskontroller **220** ist mit dem seriellen Bus **201** (entspricht **101** in Fig. 1) verbunden. Der intelligente Kommunikationskontroller **220** beinhaltet mindestens einen eigenen Protokollprozessor **221**, einen lokalen Speicher **222**, der die Datenstrukturen für die Kommunikation enthält, und eine Speicherschnittstelle **224** zum Host Computer **230**. Über die Signalleitung **223** kann der Kommunikationskontroller **220** Unterbrechungssignale an den Host Computer **230** leiten. Der Host Computer **230** verfügt normalerweise über einen internen Bus **231**, einen lokalen Prozessor **232** und einen lokalen Speicher **233**. Weiters kann er noch über Prozeß- oder Kommunikationsperipherie verfügen, die in der Fig. 2 nicht eingetragen ist, da sie für die gegenständliche Erfindung nicht wesentlich ist.

Fig. 3 zeigt die Struktur eines Sensorknoten. Er verfügt normalerweise über einen internen Bus **310** an dem ein einfacher unintelligenter Kommunikationskontroller **311**, z.B. ein standardmäßigen asynchroner UART Controller, angeschlossen ist. Dieser Kommunikationskontroller **311** sendet und empfängt Nachrichten vom seriellen Bus **301** (entspricht **101** in Fig. 1) her. Weiters verfügt der einfache Knotenrechner über eine lokale Recheneinheit (CPU) **312**, einen Speicher **313** und eine Prozeßperipherie **314**. An die Prozeßperipherie kann ein Sensor oder Aktor angeschlossen werden (siehe z.B. Sensor **111** in Fig. 1). Die Rechenheit führt sowohl Protokollaufgaben, wie auch Anwendungsberechnungen durch und steuert den Datentransfer zwischen Prozeßperipherie **314**, Speicher **313** und Kommunikationskontroller **311**. Entsprechend dem gegenwärtigen Stand der Technik kann ein Sensorknoten mittels eines am Markt erhältlichen Einchip Mikrokontroller realisiert werden (z.B., Motorola HCO5 Serie).

Fig. 4 zeigt ein Zeitdiagramm, das den Nachrichtentransport am seriellen Bus veranschaulicht. Die Abszisse **400** stellt die Zeitachse dar. In diesem Beispiel liest der Sensorknoten **110** von Fig. 1 zuerst seiner Meßwert (Zeitpunkt **410**) und sendet ihn während des *a priori* bekannten Zeitintervalls **411** zum Masterknoten (Knotenrechner **140** in Fig. 1). Sensorknoten **120** von Fig. 1 liest seinen Meßwert (Zeitpunkt **420**) später und sendet ihn während des *a priori* bekannten Zeitintervalls **421** zum Masterknoten (Knotenrechner **140** in Fig. 1). Der Masterknoten (Knotenrechner **140** in Fig. 1) beginnt die Verarbeitung nach Empfang beider Nachrichten zum Zeitpunkt **401**. Da in einem zeitgesteuerten Kommunikationssystem die Übertragungsintervalle **411** und **421** *a priori* bekannt sind, kann der Sender (oder Empfänger) die Zeitintervalle zwischen den Zeitpunkten **410** und **401** bzw. **420** und **401** in jeder Übertragungsrunde exakt berechnen. Diese Zeitintervalle werden vom Sender dazu benutzt, den Zustand der gelesenen Meßgröße zum zukünftigen Zeitpunkt **401** voraus zu berechnen.

Fig. 5 zeigt das Datenformat für Analogmeßwerte. Es wird zwischen zwei Versionen der Darstellung unterschieden, einer kurzen Version von 1 Byte Länge (**510**) und einer langen Version von zwei Byte Länge (**510** and **520**). In der kurzen Version **510** werden die analogen Meßwerte im Bereich von 0 bis 200 Einheiten kodiert, wobei 0 den Meßwert 0% und 200 den Meßwert 100% des gewählten Meßbereichs des Sensors darstellt. Alle Werte die größer sind als 200 sind Fehlermeldungen. Es besteht somit die Möglichkeit, in der kurzen Version Analogmeßdaten mit einer Genauigkeit von 0.5% des gewählten Meßbereichs zu kodieren oder eine von 56 verschiedenen Fehlermeldungen zu übertragen. In der langen Version ist das erste Byte identisch mit dem Byte der kurzen Version **510**. Das erste Halbbyte **521** des zweiten Bytes **520** beinhaltet eine Differen-

zierung des Meßwertes im ersten Byte in 16 Unterklassen, was zu einer Gesamtgenauigkeit von etwas weniger als 12 Bit führt. Das zweite Halbbyte 522 des zweiten Bytes beinhaltet einen Konfidenzindex. Der Konfidenzindex wird vom Sensor auf der Basis des gelernten Langzeitverhalten des Meßwertes und von a priori Wissen über den technischen Prozeß errechnet und teilt dem Empfänger mit, inwieweit der Empfänger dem Meßwert vertrauen kann. Ein Konfidenzindex von 0 bedeutet, daß dem Wert nicht zu trauen ist und ein Konfidenzindex von 15 bedeutet daß der Wert richtig ist. Treten z.B. spontane Meßwertänderungen auf die auf einen möglichen Sensorfehler hindeuten, so kann der Sensor diese Information dem Empfänger über den Konfidenzindex mitteilen.

Fig. 6 zeigt den Anschluß eines Sensorsystems in einer fehlertoleranten Architektur. In einer solchen Architektur ist der Masterknoten (140 in Fig. 1) repliziert 640 und 650. Jeder der beiden Masterknoten 640 und 650 besteht aus drei Subsystemen, dem Kommunikationskontrollern zum Sensorbus 641 und 651, den Hostcomputers 642 und 652, und den Kommunikationskontrollern 643 und 653 zu den replizierten Echtzeitbussen 661 und 662 entsprechend dem oben zitierten Europäischen Patent: 0 658 257. Die Schnittstellen 645 und 655 zwischen dem Sensorbuskontrollern und dem Hostcomputern ist eine gemeinsame Speicherschnittstelle. Jeder der beiden Kommunikationskontroller 641 und 651 verfügt über 2 Anschlüsse zu den Sensorbussen 601 und 602 um alle Nachrichten auf beiden Bussen empfangen zu können. Im fehlerfreien Fall ist der Masterknoten 641 der Master von Bus 601 und der Masterknoten 642 der Master von Bus 602. Im Fehlerfall, nach Ausfall eines Masterknotens, kann der noch funktionierende Masterknoten beide Busse 601 und 602 steuern. An dem Sensorbus 601 sind die Sensorknoten 611, 621 und 631 angeschlossen. An dem Sensorbus 602 sind die Sensorknoten 612, 622 und 632 angeschlossen. Das gesteuerte Objekt 600 wird jeweils von einem Paar redundanter Sensoren 611, 612 und 621, 622 und 631, 632 beobachtet.

Fig. 7 zeigt eine Nachrichtenfolge in der Initialisierungsrunde. Auf der x-Achse ist die Zeit 700 aufgetragen. Die Initialisierungsrunde beginnt mit dem speziellen Fireworks Byte 710 vom Masterknoten. Dann folgt entsprechend dem TTP/A Protokoll ein signifikant längerer Abstand als zwischen den Datenbytes (siehe Kopetz, H. (1995) TTP/A -- A Time-Triggered Protocol of Body Electronics Using Standard UARTS. Proc. SAE World Congress, Society of Automotive Engineers, SAE Technical Paper 950039. pp. 1-9). In dem gezeigten Beispiel beinhalten die ersten vier Datenbytes 720 den eindeutigen physikalischen Namen des Knoten aus einem 32 Bit Namensraum. Dann folgt die Position des Knotens in der Meßrunde 730. Im weiteren können anwendungsspezifische Initialisierungsdaten 740, wie z.B. der Meßbereich, übertragen werden.

Nachdem die einzelnen Bausteine beschrieben wurden, wird nun die Realisierung der Erfindung anhand eines Ablaufs entsprechend den Abbildungen 1-7 erklärt.

Entsprechend dem bekannten TTP/A Protokoll (siehe Kopetz, H. (1995) TTP/A -- A Time-Triggered Protocol of Body Electronics Using Standard UARTS. Proc. SAE World Congress, Society of Automotive Engineers, SAE Technical Paper 950039. pp. 1-9) ist die Kommunikation in Runden organisiert, die jeweils mit einem Fireworksbyte vom aktiven Master beginnen. Zu Beginn einer Runde sendet der aktive Masterknoten 140 ein Feuerwerksbyte mit dem ausgewählten Rundennamen als Inhalt. Die erste Runde ist eine Initialisierungsrunde (Fig. 7), während der vom Masterknoten 140 die Parameter der Sensorknoten, wie z.B., die Position des Sensorknotens in der Meßrunde und der Meßbereich der Sensorknoten konfiguriert werden. Nach jedem Neustart eines Knotens wird der Knoten durch eine Initialisierungsrunde vom Master parametrisiert. Durch diese Eigenschaft der Erfindung ist es möglich, die Software im Sensorknoten anwendungsunabhängig auszulegen, in ROM Speicher abzulegen, und Sensorknoten kostengünstig in einer Massenproduktion nur mit einer individuellen Seriennummer (physikalischer Name des Knotens) herzustellen.

Wenn alle Sensorknoten initialisiert sind folgt ein Fireworksbyte, das den Beginn einer Meßrunde ankündigt. Eine Meßrunde ist a priori so aufgebaut, daß der Meßwert mit der geringsten Dynamik 410 zuerst, der Meßwert mit der größten Dynamik 420 zuletzt gesendet werden. Da den Sensorknoten die Dauer einer Meßrunde a priori bekannt ist, können sie aufgrund der Langzeitbeobachtung des Meßwertes und des bekannten aufgrund der kompakten Datenkodierung (510 bzw. 520) kurzen Zeitintervalle 410 bis 401 bzw. 420 bis 401, eine "State Estimation" in die Zukunft bis zum Zeitpunkt der Verarbeitung 401 beim Empfänger durchführen. Eine solche "State

Estimation" wird im einfachsten Fall über die bekannte Taylor Reihenentwicklung um den Meßpunkt

$$f(t. \text{Empfänger}) = f(t. \text{Meßpunkt}) + (t. \text{Empfänger} - t. \text{Meßpunkt}) \frac{df(t. \text{Meßpunkt})}{dt}$$

5 durchgeführt. Im allgemeinen Fall wird jeder Sensorknoten zu einem unterschiedlichen Zeitpunkt (t.Meßpunkt) messen, aber den gleichen Empfangszeitpunkt (t.Empfänger) verwenden, um zu gewährleisten, daß zu Beginn der Verarbeitung beim Empfänger alle Meßwerte zeitlich und räumlich konsistent sind. Es ist natürlich möglich, komplexere State Estimation Verfahren in den  
10 Sensorknoten zu realisieren. Der Kommunikationskontroller des empfangende Masterknoten 220 verfügt über einen Zwischenspeicher 222 indem alle Sensorwerte bis zum Empfang des letzten Sensorwertes einer Runde zwischengespeichert werden, so daß eine Menge von räumlich und zeitlich konsistenten Meßwerten nahezu gleichzeitig dem Host Computer 230 des Masterknotens über die Speicherschnittstelle 224 zur Verfügung gestellt werden kann.

15 Die *a priori* bekannten Abstände des Firework Bytes, generiert vom Master 140, können auch genutzt werden, um die Uhren der Sensorknoten nach bekannten Verfahren zu resynchronisieren, da der Masterknoten normalerweise über einen Oszillator von höherer Ganggenauigkeit verfügen wird wie die Sensorknoten.

20 In einer fehlertoleranten Konfiguration (Fig. 6) empfängt jeder der replizierten Masterknoten 640 und 650 alle Werte der replizierten Sensoren über die redundanten Sensorbusse 601 und 602. Jeder der beiden Kontroller 641 und 651 berechnet auf der Basis dieser replizierten Sensorwerte und der damit verbundenen anwendungsspezifischen Konfidenzindexes 522 den gleichen einheitlichen Meßwert, der über die Speicherschnittstellen 645 und 655 den Host Computern 642 und 652 angeboten wird. Ein Beispiel für eine solche einheitlichen Meßwertbildung ist die Bildung des mit  
25 dem Konfidenzindex gewichteten Mittelwerts aus einer Menge von Meßwerten. Wenn ein Sensor oder ein Bus ausfällt, so berechnen die Kontroller 641 und 651 aus den verbleibenden Sensorwerten den gewünschten Meßwert, einschließlich eines neuen Konfidenzindexes 522. Wenn ein Master ausfällt, so übernimmt der verbleibende Master die Steuerung beider Busse. Den Hostcomputer 642 bzw. 652 im Masterknoten bleibt die Existenz der Fehlertoleranz im Sensorsystem  
30 somit verborgen. In dieser Architektur führt die Fehlertoleranz des Sensorsystems zu keiner Erhöhung der Komplexität der Software in den Hostcomputern 642 und 652, was eine wichtige Eigenschaft dieser Erfindung darstellt. Die Problematik einer hohen Softwarekomplexität ist gegenwärtig ein ernstzunehmendes industrielles Problem.

35 In einer fehlertoleranten Konfiguration können auch drei oder mehr Busse mit drei oder mehr Masterknoten konfiguriert werden, so daß ein höherer Grad der Fehlertoleranz erreicht wird.

#### PATENTANSPRÜCHE:

- 40 1. Methode zur Übertragung von Nachrichten mit zeitabhängigem Inhalt in einem zeitgesteuerten verteilten Computersystem, in dem eine Vielzahl von Knotenrechnern über einen oder mehrere Broadcast Kommunikationskanäle verbunden sind und wo jeder Knotenrechner über eine Kommunikationskontrolleinheit mit den entsprechenden Anschlüssen an  
45 die Kommunikationskanäle verfügt und wo der Zugriff des Senders auf die Kommunikationskanäle entsprechend einem zyklischen Zeitscheibenverfahren erfolgt, **dadurch gekennzeichnet, daß** das dem Sender *a priori* bekannte Zeitintervall bis zum Beginn der Bearbeitung einer Nachricht beim Empfänger vom Sender dazu verwendet wird, den zeitabhängigen Inhalt der Nachricht derart zu aktualisieren, daß der Inhalt der Nachricht zum Zeitpunkt des Eintreffens der Nachricht beim Empfänger oder zum Zeitpunkt des Beginns  
50 der Bearbeitung der Nachricht durch den Empfänger ein weitgehend zeitrichtiges Abbild der durch die Nachricht zu vermittelnden sich nach bekannten Regeln ändernden Wirklichkeit wiedergibt.
- 55 2. Kommunikationsmethode zur Übertragung von zeitabhängigen Nachrichteninhalten **dadurch gekennzeichnet, daß** ein Analogwert eines Sensors in einem einzigen Byte dargestellt ist, wobei der dimensionslose Wertebereich 0-100% des Analogwertes in den

Codeworten 0 - 199 codiert wird und wo die verbleibenden 56 Codeworte zur Übertragung von Fehlermeldungen vom Sensor an den Empfänger verwendet werden.

3. Kommunikationsmethode nach Anspruch 2 zur Übertragung von zeitabhängigen Nachrichteninhalten **dadurch gekennzeichnet, daß** ein Analogwert eines Sensors in zwei Byte dargestellt ist, wobei das erste Byte entsprechend Anspruch 2 aufgebaut ist und wo die erste Hälfte des zweiten Bytes eine weitergehende Differenzierung des Wertes im ersten Byte in 16 Klassen angibt, und die zweite Hälfte des zweiten Bytes einen Konfidenzindex von 0 bis 15 enthält, wobei der Konfidenzindex 0 bedeutet, daß dem Meßwert nicht zu trauen ist.
4. Kommunikationsmethode **dadurch gekennzeichnet, daß** die Übertragung einer zusammengehörigen Menge von Nachrichten an einen Empfänger *a priori* derart festgelegt wird, daß die Nachrichten entsprechend der Dynamik des Nachrichteninhaltes sortiert werden und die Nachricht mit der geringsten Dynamik zuerst und die Nachricht mit der größten Dynamik zuletzt gesendet wird.
5. Kommunikationsmethode **dadurch gekennzeichnet, daß** die Sensorknoten die ihnen *a priori* bekannten Zeitpunkte des Eintreffens von *a priori* definierten Nachrichten vom aktiven Masterknoten dazu verwenden, die lokalen Uhren der Empfänger zu resynchronisieren.
6. Kommunikationsmethode **dadurch gekennzeichnet, daß** in einem oder mehreren Masterknoten die Vereinheitlichung von über replizierte Bussen und über replizierte Sensoren erfaßten redundanten Meßwerte vorgenommen wird.
7. Kommunikationsmethode, **dadurch gekennzeichnet, daß** die Sensorknoten nach jedem Einschalten im Rahmen einer oder mehrere Initialisierungsrunden durch den Masterknoten neu parametrisiert werden, um die Software im Sensorknoten anwendungsunabhängig auslegen zu können.
8. Knotenrechner zur Erfassung und Bearbeiten von Nachrichten mit zeitabhängigem Inhalt in einem zeitgesteuerten verteilten Computersystem, in dem eine Vielzahl von Knotenrechnern über einen oder mehrere Broadcast Kommunikationskanäle verbunden sind und wo jeder Knotenrechner über mindestens eine Kommunikationskontrolleinheit mit den entsprechenden Anschlüssen an die Kommunikationskanäle verfügt und wo der Zugriff des Senders auf die Kommunikationskanäle entsprechend einem zyklischen Zeitscheibenverfahren erfolgt, **dadurch gekennzeichnet, daß** das dem sendenden Knotenrechner *a priori* bekannte Zeitintervall bis zum Beginn der Bearbeitung einer Nachricht beim Empfänger vom sendenden Knotenrechner dazu verwendet wird, den zeitabhängigen Inhalt der Nachricht so zu aktualisieren, daß der Inhalt der Nachricht zum Zeitpunkt des Beginns der Bearbeitung der Nachricht durch den Empfänger oder zum Zeitpunkt des Eintreffens der Nachricht beim Empfänger ein weitgehend zeitrichtiges Abbild der durch die Nachricht zu vermittelnden sich nach bekannten Regeln ändernden Wirklichkeit wiedergibt.
9. Knotenrechner zur Erfassung und Bearbeiten von Nachrichten mit zeitabhängigem Inhalt in einem zeitgesteuerten verteilten Computersystem, in dem eine Vielzahl von Knotenrechnern über einen oder mehrere Broadcast Kommunikationskanäle verbunden sind und wo jeder Knotenrechner über mindestens eine Kommunikationskontrolleinheit mit den entsprechenden Anschlüssen an die Kommunikationskanäle verfügt und wo der Zugriff des Senders auf die Kommunikationskanäle entsprechend einem zyklischen Zeitscheibenverfahren erfolgt, **dadurch gekennzeichnet, daß** das der Knotenrechner analoge Sensorwerte entsprechend den in Ansprüchen (2) oder (3) festgelegten Datenformaten kodiert und dekodiert.
10. Sensorknoten zur Erfassung und Bearbeiten von Meßdaten mit zeitabhängigem Inhalt in einem zeitgesteuerten verteilten Computersystem **dadurch gekennzeichnet, daß** die Sensorknoten nach jedem Einschalten im Rahmen einer oder mehrere Initialisierungsrunden durch den Masterknoten neu parametrisiert werden, um die Produktion der Sensorknoten anwendungsunabhängig auslegen zu können.
11. Masterknoten zur Erfassung und Bearbeiten von Nachrichten mit zeitabhängigem Inhalt in einem zeitgesteuerten verteilten Computersystem **dadurch gekennzeichnet, daß** die Kommunikationskontrolleinheit des Masterknotens eine *a priori* bekannte zusammengehörigen Menge von Nachrichten zwischenspeichert bis die letzte Nachricht dieser

Menge eingetroffen ist, und wo nach Eintreffen dieser letzten Nachricht alle Nachrichten nahezu gleichzeitig dem Host in diesem Masterknoten zur Verfügung gestellt werden.

12. Masterknoten zur Erfassung und Bearbeiten von Meßdaten mit zeitabhängigem Inhalt in einem zeitgesteuerten verteilten Computersystem **dadurch gekennzeichnet, daß** die Kommunikationskontroller der Masterknoten autonom die Abgleichung der von unterschiedlichen replizierten Sensorknoten erfaßten Meßdaten vornehmen und die abgeglichenen Meßdaten in einheitlicher Form nahezu gleichzeitig den Host Computern zur Verfügung stellen.

5

10

**HIEZU 3 BLATT ZEICHNUNGEN**

15

20

25

30

35

40

45

50

55

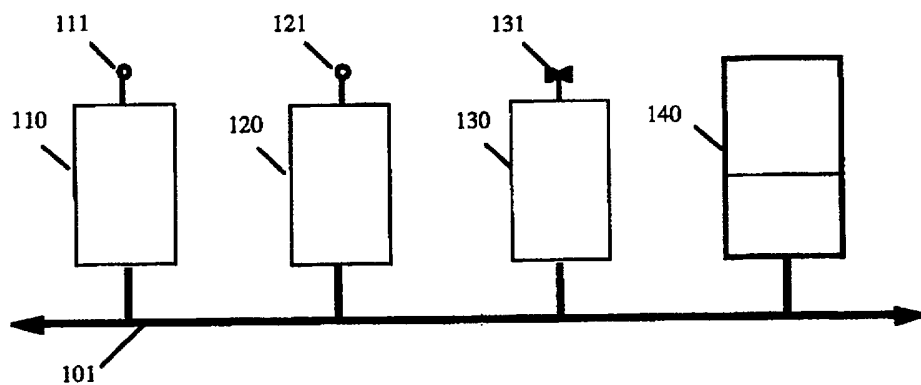


FIG. 1

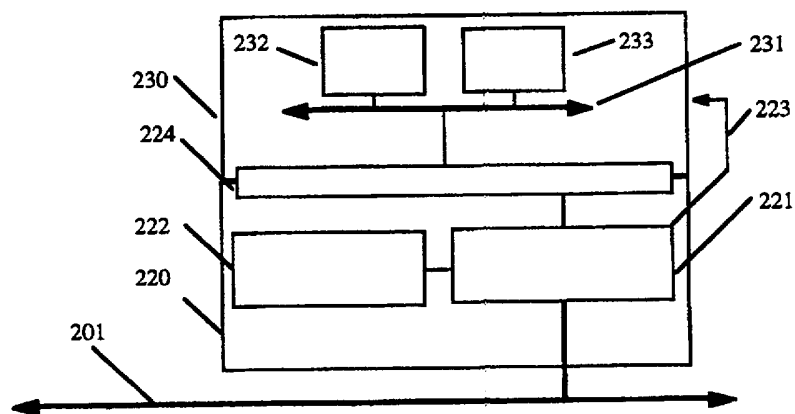


FIG. 2

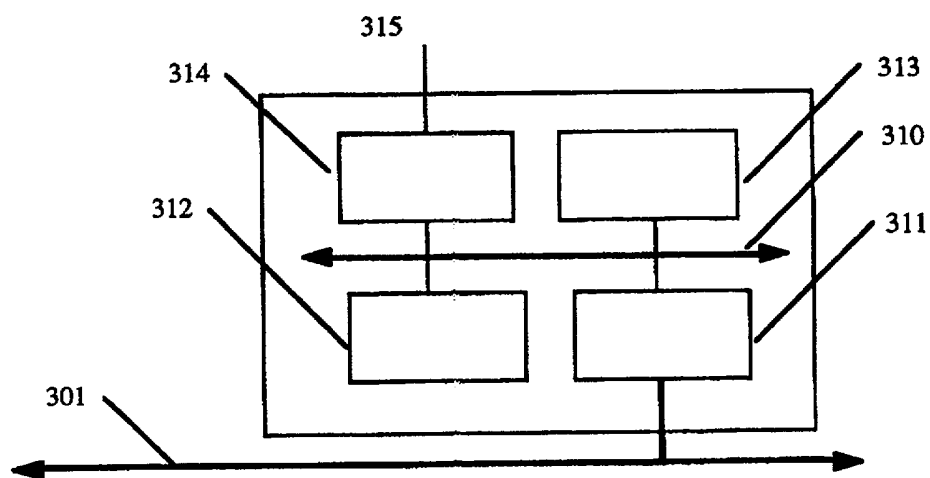


FIG. 3

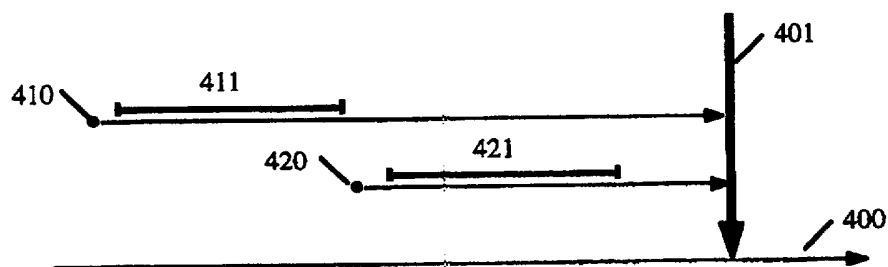


FIG. 4

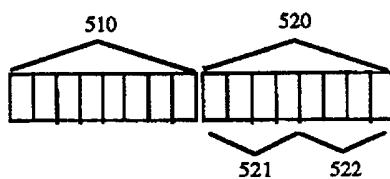


FIG. 5

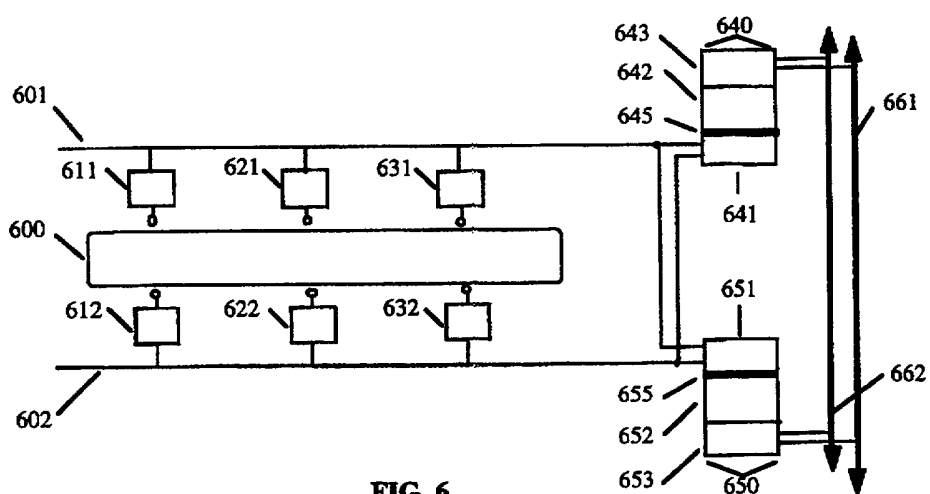


FIG. 6

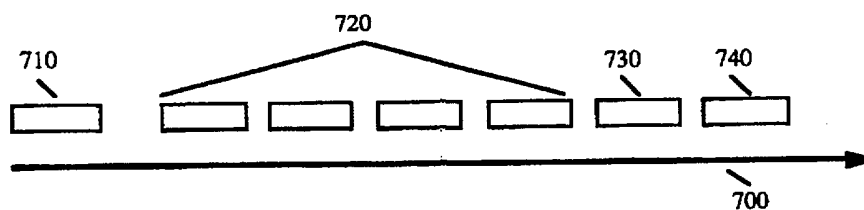


FIG. 7