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

(10) **DE 600 04 500 T2** 2004.02.26

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 1 163 583 B1**
(21) Deutsches Aktenzeichen: **600 04 500.5**
(86) PCT-Aktenzeichen: **PCT/SE00/00325**
(96) Europäisches Aktenzeichen: **00 911 531.2**
(87) PCT-Veröffentlichungs-Nr.: **WO 00/50995**
(86) PCT-Anmeldetag: **18.02.2000**
(87) Veröffentlichungstag der PCT-Anmeldung: **31.08.2000**
(97) Erstveröffentlichung durch das EPA: **19.12.2001**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **13.08.2003**
(47) Veröffentlichungstag im Patentblatt: **26.02.2004**

(51) Int Cl.[7]: **G06F 11/16**
H04Q 7/34

(30) Unionspriorität:
257233 **25.02.1999** **US**

(84) Benannte Vertragsstaaten:
AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LI, LU, MC, NL, PT, SE

(73) Patentinhaber:
Telefonaktiebolaget L M Ericsson (publ), Stockholm, SE

(72) Erfinder:
KARLSSON, Marcus, S-972 41 Lulea, SE;
RYNBÄCK, Patrik, S-977 51 Lulea, SE

(74) Vertreter:
HOFFMANN · EITLE, 81925 München

(54) Bezeichnung: **ZUSTANDSSYNCHRONISATION IN REDUNDANTEN SYSTEMEN**

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

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

Beschreibung

1. GEBIET DER ERFINDUNG

[0001] Die vorliegende Erfindung steht in Verbindung mit Synchronisation von einem redundanten System, und insbesondere mit Synchronisation von einem redundanten Computersystem, welches zwei unabhängige Einheiten hat, welche simultan die selben Aufgaben ausführen.

2. STAND DER TECHNIK UND WEITERE GESICHTSPUNKTE

[0002] Einige Aktivitäten sind so außergewöhnlich wichtig, dass es verständlich ist, zwei Einheiten oder Prozessoren zum Durchführen der Aktivitäten zu haben, nur für den Fall, dass eine der Einheiten versagt oder abbricht (beispielsweise abstürzt). Es. wird angenommen, dass dies beispielsweise in zukünftigen Generationen von mobilen Telekommunikationssystemen der Fall sein wird. Es wird Bezug genommen auf US-Patent 5,469,503 von Butensky et al., welchen einen Wiederherstellungsalgorithmus für ein redundantes System in einer Telekommunikationsumgebung offenbart.

[0003] Bei solch kritischen Einsatzsituationen werden mehrere Einheiten (beispielsweise Computerprozessoren oder CPU's) verwendet. In einigen Fällen übernimmt eine oder übernehmen mehrere der Einheiten die Rolle einer Standby-Einheit, welche im wesentlichen im Leerlauf ist, bis zu der Zeit, da es notwendig ist eine aktive Einheit zu ersetzen. In anderen Fällen führen mehrere redundante Einheiten die selben Aufgaben simultan aber unabhängig aus. Unglücklicherweise können bei diesen anderen Fällen die redundanten Einheiten unter bestimmten Umständen gegenseitig aus der Phase geraten. Unter solchen Umständen ist es notwendig, die redundanten Einheiten wieder aufeinander abzustimmen oder zu „resynchronisieren".

[0004] Eine Abstimmung von redundanten Einheiten wurde hier zuvor unter Verwendung unterschiedlicher Techniken zugewiesen. Eine bekannte Technik besteht darin, dass die zwei redundanten Systeme, beispielsweise zwei CPU's, einen selben Takt und selben Speicher verwenden. Bei dieser Technik führen die zwei CPU's dieselben Instruktionen aus demselben Speicher zur selben Zeit aus, wodurch die Notwendigkeit zur Software-Synchronisation vermieden wird. Eine dritte Einheit vergleicht die Ergebnisse von den zwei CPU's. Ein Nachteil dieser Technik besteht darin, dass deren Implementierung spezielle Hardware anstatt herkömmlicher Komponenten benötigt.

[0005] Eine Verwendung von einer dritten Einheit, welche eine Synchronisation von zwei Prozessoren abstimmt, wird im US-Patent 5,748,873 von Ohguro et al. gelehrt. Wenn ein erster Prozessor mit einem zweiten Prozessor resynchronisieren muss, sendet der erste Prozessor eine Resynchronisationsanzeige an eine Übereinstimmung-Steuerungslogik, welche dann synchrone Neustart-Anzeigesignale ausgibt, so dass sowohl der erste als auch der zweite Prozessor synchron neugestartet werden.

[0006] Einige redundante Einheiten (welche unabhängig dieselbe Aufgabe ausführen), enthalten Zustandsmaschinen, welche auf der Basis von externer Anregung synchronisiert werden, wie zum Beispiel Busereignissen. Wenn jedoch eine der redundanten Einheiten neugestartet wird, (beispielsweise nach einem Versagen oder dergleichen erneut gebootet wird), hat die Neustarteinheit irgendwie die andere Einheit einzuholen, so dass die Aufgaben und jegliche Zustandsmaschinen von der Neustarteinheit synchronisiert werden können. Eine mögliche Lösung für dieses Synchronisationsproblem liegt im Ignorieren externer Anregung und einer Übertragung aller Zustände von einer aktiven Einheit an die jüngst neugestartete Einheit. Unglücklicherweise würde diese Lösung erfordern, dass beide Einheiten zumindest temporär während der Übertragung von Zuständen für mit den zwei Einheiten kommunizierende externe Einrichtungen isoliert erscheinen. Wenn das Gesamtsystem viel Zustandsinformation enthält, könnte eine solche Resynchronisationszeit inakzeptabel lang sein.

[0007] Was daher benötigt wird, und eine Aufgabe von der vorliegenden Erfindung ist, ist ein wirksamer Weg zur Resynchronisation redundanter Einheiten, welche synchronisierte Zustandsmaschinen verwenden.

KURZE ZUSAMMENFASSUNG VON DER ERFINDUNG

[0008] Ein redundantes System enthält eine erste Einheit, eine zweite Einheit und einen Synchronisationsserver. Die erste Einheit und eine zweite Einheit enthalten jeweils mehrere Zustandsmaschinen zur Durchführung von Aufgaben. Zumindest einige der Zustandsmaschinen, welche an der ersten Einheit bereitgestellt sind, führen gleichzeitig die selben Aufgaben durch, wie zumindest einige entsprechende der Zustandsmaschinen, welche an der zweiten Einheit bereitgestellt sind.

[0009] Wenn eine Zustandsmaschine einer zweiten Einheit bestimmt, dass sie resynchronisiert werden muss, empfängt der Synchronisationsserver gemäß der Erfindung eine Resynchronisationsanforderung von der Zustandsmaschine, welche eine Resynchronisation anfordert, und stellt darauf eine out-of-synchronization Anzeige an einer entsprechenden Zustandsmaschine in der ersten Einheit bereit. In Ansprechen auf die out-of-synchronization Anzeige von dem Synchronisationsserver erzeugt die entsprechende Zustandsmaschine in der ersten Einheit eine Resynchronisationsanforderung, zu einer Zeit, welche durch die entsprechende Zustandsmaschine in der ersten Einheit als geeignet erachtet wird: In An-

sprechen auf die Resynchronisationsanforderung von der entsprechenden Zustandsmaschine in der ersten Einheit gibt der Synchronisationsserver einen Resynchronisationsbefehl sowohl an die entsprechende Zustandsmaschine in der ersten Einheit, als auch an die anfordernde Zustandsmaschine von der zweiten Einheit aus, welche zur Resynchronisation nachgesucht hat. Die Zustandsmaschinen der ersten und zweiten Einheit resynchronisieren gleichzeitig in Ansprechen auf den Resynchronisationsbefehl.

[0010] Die erste Einheit und/oder die zweite Einheit enthält eine Neustart-Logik, welche eine Neustart-Reihenfolge der Mehrzahl von Zustandsmaschinen vorschreibt, wenn eine Einheit neugestartet oder neugebootet wird. Dies stellt eine graduelle Anregung der Zustandsmaschinen der redundanten Einheiten bereit. Vorzugsweise basiert die vorgeschriebene Reihenfolge auf einer Zustandsmaschine zu einem Zeitpunkt. Wann eine Zustandsmaschine neugestartet wird, hängt von der einzelnen Zustandsmaschine ab, wann sie gemäß den oben zusammengefassten Schritten synchronisiert.

[0011] Bei einer dargestellten Ausführungsform befinden sich die erste Einheit und die zweite Einheit in einem Knoten eines zellularen Telekommunikationssystems, wie beispielsweise einem Basisstation(BTS)-Knoten.

[0012] Der Synchronisationsserver kann eine verschiedenartige Einheit sein oder kann in einer der ersten Einheit und der zweiten Einheit enthalten sein.

[0013] Diese Aufgaben werden auf vorteilhafte Weise gelöst, indem grundlegend Merkmale verwendet werden, welche in den unabhängigen Ansprüchen dargelegt sind. Weitere Verbesserungen werden durch die Unteransprüche bereitgestellt.

KURZE BESCHREIBUNG DER ZEICHNUNG

[0014] Das Vorhergehende und weitere Aufgaben, Merkmale und Vorteile der Erfindung werden ersichtlich durch die folgende speziellere Beschreibung von bevorzugten Ausführungsformen, wie sie in den begleitenden Zeichnungen dargestellt sind, in denen sich Bezugszeichen über die gesamten Ansichten hinweg auf dieselben Teile beziehen. Es ist nicht notwendig, die Zeichnungen zu skalieren, anstelle dessen werden Hervorhebungen über die Darstellung der Prinzipien der Erfindung gesetzt.

[0015] **Fig.** 1 ist eine schematische Ansicht eines redundanten Systems gemäß einer ersten Ausführungsform der Erfindung.

[0016] **Fig.** 1A ist eine schematische Ansicht eines redundanten Systems gemäß einer anderen Ausführungsform der Erfindung.

[0017] **Fig.** 1B ist eine schematische Ansicht eines redundanten Systems gemäß einer anderen Ausführungsform der Erfindung, bei welcher sich eine Neustart-Logik in einem Knoten von einem Synchronisationsserver befindet.

[0018] **Fig.** 1C ist eine schematische Ansicht eines redundanten Systems gemäß einer anderen Ausführungsform der Erfindung, bei welcher sich eine Neustart-Logik in einem Knoten befindet, welcher unterschiedlich ist von einem Knoten der ersten Einheit, dem Knoten der zweiten Einheit oder dem Knoten des Synchronisationsservers.

[0019] **Fig.** 2 ist eine schematische Ansicht, welche ein Meldungsschema zur Resynchronisation von einer Zustandsmaschine für die Ausführungsform von **Fig.** 1 zeigt.

[0020] **Fig.** 3 ist eine schematische Ansicht, welche einen Neustartablauf gemäß eines Modus der vorliegenden Erfindung darstellt.

[0021] **Fig.** 3A ist eine schematische Ansicht, welche einen Neustartablauf eines anderen Modus der vorliegenden Erfindung darstellt.

[0022] **Fig.** 4 ist eine schematische Ansicht, welche eine Implementierung der Ausführungsform von **Fig.** 1 in einem Knoten eines zellularen Telekommunikationssystems zeigt.

DETAILLIERTE BESCHREIBUNG DER ZEICHNUNGEN.

[0023] Bei der folgenden Beschreibung werden aus Gründen der Erläuterung und nicht der Beschränkung spezifische Details dargelegt, beispielsweise bestimmte Architekturen, Schnittstellen, Techniken, usw., um ein vollständiges Verständnis von der vorliegenden Erfindung bereitzustellen. Jedoch wird es dem Fachmann klar sein, dass die vorliegende Erfindung in anderen Ausführungsformen praktiziert werden kann, welche von diesen spezifischen Details abweichen. In anderen Fällen werden detaillierte Beschreibungen von bekannten Vorrichtungen, Schaltungen und Verfahren ausgelassen, um die Beschreibung der vorliegenden Erfindung nicht mit unnötigen Details zu verkomplizieren.

[0024] **Fig.** 1 zeigt ein redundantes System **20** gemäß einer ersten Ausführungsform der Erfindung. Das redundante System **20** enthält eine erste und zweite Einheit, dargestellt als erste CPU **21** und zweite CPU **22**; als auch einen Resynchronisationsserver **24**. Jede der ersten CPU **21** und zweiten CPU **22** ist vorzugsweise eine Verarbeitungseinheit, beispielsweise ein programmierter Mikroprozessor oder dergleichen. Jede der ersten CPU **21** und zweiten CPU **22** enthält mehrere Software-Zustandsmaschinen. Wie in **Fig.** 1 gezeigt, hat erste CPU **21** Zustandsmaschinen $SM_{1,A}$, $SM_{1,B}$, ... $SM_{1,N}$, während zweite CPU **22** Zustandsmaschinen $SM_{2,A}$, $SM_{2,B}$, ... $SM_{2,N}$ hat. Da System **20** ein redundantes System mit erster CPU **21** und zweiter CPU **22** ist, welche unabhängig aber (unter normalen Umständen) simultan dieselben Aufgaben ausführen, führt Zustandsmaschine $SM_{1,A}$ von erster CPU **21** dieselbe Aufgabe von Zustandsmaschine $SM_{2,A}$ von zweiter CPU **22** durch; Zustandsmaschine $SM_{1,B}$ von erster CPU **21** führt dieselbe Aufgabe von Zustandsmaschine $SM_{2,B}$ von zweiter CPU **22** durch; usw., gemäß einem alphabetischen

Index-Muster (subscripted pattern). Somit hat die Redundanz von System **20** die Zustandsmaschinen von **Fig.** 1 in Paaren bereitgestellt, einem ersten Paar $SM_{1,A}$ und $SM_{1,B}$; einem zweiten Paar $SM_{2,A}$ und $SM_{2,B}$; usw., fortführend bis zu einem n-ten Paar $SM_{1,N}$ und $SM_{1,N}$.

[0025] Vorzugsweise wird ein oder mehr der Paare von Zustandsmaschinen, welche in **Fig.** 1 gezeigt sind, auf der Basis von externer Anregung synchronisiert, beispielsweise Bus-Ereignisse. **Fig.** 1 stellt dar, dass eine Anregung A gleichzeitig an Zustandsmaschine $SM_{1,A}$ und Zustandsmaschine $SM_{2,A}$ angelegt wird; eine Anregung B wird gleichzeitig an Zustandsmaschine $SM_{1,B}$ und Zustandsmaschine $SM_{2,B}$ angelegt; usw. Beim normalen Betrieb werden die Zustandsmaschinen von jedem Paar in Synchronisation aufrecht erhalten, aufgrund der Tatsache, dass beide Zustandsmaschinen von dem Paar gleichzeitig dieselbe Anregung empfangen.

[0026] Jede der ersten CPU **21** und zweiten CPU **22** enthält ebenfalls eine Neustart-Logik. In **Fig.** 1 ist die erste CPU **21** so dargestellt, dass sie Neustart-Logik $\mathbf{30_1}$ hat, während die zweite CPU so dargestellt ist, dass sie Neustart-Logik $\mathbf{30_2}$ hat.

[0027] Bei der in **Fig.** 1 gezeigten bestimmten Ausführungsform ist Resynchronisationsserver **24** eine separate Einheit, welche von erster CPU **21** und zweiter CPU **22** verschieden ist, jedoch sowohl mit erster CPU **21**, als auch zweiter CPU **22** über ein. Bussystem **34** verbunden ist. In **Fig.** 1A ist im Gegensatz dazu Resynchronisationsserver **24'** in einer zweiten CPU **22'** enthalten. Ein solches Einbeziehen von Resynchronisationsserver **24'** mit zweiter CPU **22'** ist der hauptsächliche Unterschied zwischen den Ausführungsformen von **Fig.** 1 und **Fig.** 1A, es wird verständlich sein, dass die entsprechenden Elemente der Ausführungsform von **Fig.** 1A mit strichindexierten Bezugsziffern, relativ zu den Gegenstücken von **Fig.** 1, gezeigt werden. Die Betriebsprinzipien der Erfindung, wie hier weiter beschrieben, werden als gleichbedeutend anwendbar auf beide Ausführungsformen verstanden.

[0028] **Fig.** 2 zeigt ein grundlegendes Meldungsschema, welches eine Resynchronisation im Kontext der Ausführungsform von **Fig.** 1 ermöglicht. In dem in **Fig.** 2 dargestellten Szenario wird angenommen, dass Zustandsmaschine $SM_{2,B}$ aus der Synchronisation ist, wie durch Zustandsmaschine $SM_{2,B}$ oder anders bestimmt. In einem solchen Fall zeigt **Fig.** 2, dass Zustandsmaschine $SM_{2,B}$ eine Resynchronisationsanforderung **2–1** auf Bussystem **34** an Resynchronisationsserver **24** sendet. Nach Empfangen der Resynchronisationsanforderung **2–1** stellt Resynchronisationsserver **24** eine out-of-synchronization Anzeige **2–2** an der entsprechenden Zustandsmaschine in CPU **21** bereit, d. h. Zustandsmaschine $SM_{1,B}$. Dann erzeugt, zu einer durch Zustandsmaschine $SM_{1,B}$ bestimmten geeigneten Zeit, und in Ansprechen auf die out-of-synchronization Anzeige **2–2** vom Resynchronisationsserver **24**, Zustandsmaschine $SM_{1,B}$ von erster CPU **21** ihre eigene Resynchronisationsanforderung **2–3**, welche über Bussystem **34** an Resynchronisationsserver **24** gesendet wird. Dann, nach Empfang sowohl von der Resynchronisationsanforderung **2–1** von Zustandsmaschine $SM_{2,B}$ als auch der Resynchronisationsanforderung **2–3** von Zustandsmaschine $SM_{1,B}$, gibt Resynchronisationsserver **24** einen Resynchronisationsbefehl **2–4** aus, welcher sowohl an Zustandsmaschine $SM_{1,B}$ von erster CPU **21**, als auch an Zustandsmaschine $SM_{2,B}$ von zweiter CPU **22** angelegt wird. Nach Empfang des Resynchronisationsbefehls **2–4** werden sowohl Zustandsmaschine $SM_{1,B}$ als auch Zustandsmaschine $SM_{2,B}$ neugestartet. Nach Neustart von Zustandsmaschine $SM_{1,B}$ und Zustandsmaschine $SM_{2,B}$ wird beim Empfang von beiden Zustandsmaschinen durch Busereignisse von Anregung B, eine Synchronisation erreicht.

[0029] Gemäß eines Modus der vorliegenden Erfindung, wie in **Fig.** 3 dargestellt, hat, wenn entweder erste CPU **21** oder zweite CPU **22** neugestartet oder neugebootet werden müssen (beispielsweise nach einem Systemcrash), die Neustart-Logik von jeder CPU eine vorgeschriebene Reihenfolge zur Initialisierung und Ausführung der mehreren Zustandsmaschinen. Ein Beispiel einer solchen vorgeschriebenen Reihenfolge ist in **Fig.** 3 dargestellt, wobei eine repräsentative Neustart-Logik (Neustart-Logik $\mathbf{30_2}$) mit vorgeschriebener Neustart-Zeitreihenfolge der Zustandsmaschinen für zweite CPU **22** gezeigt ist. In dem Beispiel von **Fig.** 3 ist zu sehen, dass Neustart-Logik $\mathbf{30_2}$ zunächst Zustandsmaschine $SM_{2,A}$ (angezeigt durch Aktion **3-1**) vor Starten von Zustandsmaschine $SM_{2,B}$ startet, und dass nach Starten von Zustandsmaschine $SM_{2,B}$ (angezeigt durch Aktion 3– 2) die Neustart- und Resynchronisationslogik $\mathbf{30_2}$ eine vorbestimmte Antwort oder Bestätigung empfangen muß (angezeigt durch Aktion **3–3**), bevor eine jegliche andere Zustandsmaschine gestartet wird. **Fig.** 3 zeigt, dass Zustandsmaschine $SM_{2,N}$ durch Aktion 3–q gestartet wird. Somit geschieht es bei dem hier dargestellten bestimmten Modus, dass die mehreren Zustandsmaschinen einer CPU jeweils auf Basis von einer Zustandsmaschine neugestartet werden.

[0030] Bei einem anderen in **Fig.** 3A dargestellten Modus der Erfindung bestimmt, wenn entweder erste CPU **21** oder zweite CPU **22** neugestartet oder neugebootet werden muß, jede Zustandsmaschine von der neubootenden CPU für sich selber, wann sie synchronisieren wird. In dem in **Fig.** 3A dargestellten bestimmten Szenario startet Neustart-Steuerlogik 30_2 von zweiter CPU **22** simultan jeweils jede der Zustandsmaschinen $SM_{2,A}$, $SM_{2,B}$ und $SM_{2,N}$ zur Zeit T1, wie durch Pfeile **3A–1** bis **3A–N** angezeigt. Danach bestimmt jede Zustandsmaschine, wann sie ihren eigenen Resynchronisationsprozeß beginnen wird. Beispielsweise bestimmt Zustandsmaschine $SM_{2,A}$, dass sie ihre Resynchronisation zur Zeit T2 beginnen wird; Zustandsmaschine $SM_{2,B}$ bestimmt, dass sie

ihre Resynchronisation zur Zeit T3 beginnen wird; Zustandsmaschine $SM_{2,N}$ bestimmt, dass sie ihre Resynchronisation zur Zeit T4 beginnen wird.

[0031] Die folgende Diskussion des Resynchronisationsprozeß ist sowohl auf den Modus von **Fig.** 3 als auch von **Fig.** 3A, und Variationen davon, anwendbar. Da irgendeine der Zustandsmaschinen von einer CPU ihre Resynchronisation beginnt, sendet die neugestartete Zustandsmaschine die Resynchronisationsanforderung **2–1** von **Fig.** 2 an Resynchronisationsserver **24**, und die Aktivitäten von **Fig.** 2 werden relativ zu der neugestarteten Zustandsmaschine und ihrer entsprechenden oder gepaarten Zustandsmaschine in der anderen CPU durchgeführt. Somit werden in der Situation eines CPU-Neustarts die in **Fig.** 2 dargestellten Resynchronisationsprozeduren oder Aktivitäten für jede Zustandsmaschine in der neugestarteten CPU durchgeführt, welche einer Resynchronisation bedarf.

[0032] Nach Empfang der out-of-synchronization Anzeige **2-2** hängt es von der entsprechenden Zustandsmaschine in der anderen CPU davon ab, ob die entsprechende Zustandsmaschine in der anderen CPU in der Lage zur Resynchronisation ist. In diesem Zusammenhang, und wie oben dargelegt, wurde in **Fig.** 2 zur Kenntnis genommen, dass die Zustandsmaschine von erster CPU **21** ihre eigene Resynchronisationsanforderung **2-3** zu einer Zeit erzeugt, welche von der Zustandsmaschine von erster CPU **21** als geeignet erscheint. In diesem Zusammenhang, wenn die Zustandsmaschine der ersten CPU **21** eine wichtige Aktivität zu der Zeit durchführt, bei der sie die out-of-synchronization Anzeige **2–2** empfängt, hört die Zustandsmaschine von erster CPU **21** aus Rücksicht gegen die out-of-synchronization Anzeige **2–2** nicht mit dieser wichtigen Aktivität auf. Vielmehr bestimmt die Zustandsmaschine von erster CPU **21** für sich selber, wann sie eine zufriedenstellende Gelegenheit zur Resynchronisation hat, beispielsweise wenn derzeit keine kritischen Aktivitäten durchgeführt werden oder kritische externe Anregungen Aufmerksamkeit bedürfen. Was als „kritisch" bestimmt wird, kann durch die Züstandsmaschine von erster CPU **21** in Übereinstimmung mit vorbestimmten Prioritäten festgesetzt werden. Somit erzeugt die Zustandsmaschine von erster CPU **21** eine Resynchronisationsanforderung nur nachdem die Zustandsmaschine von erster CPU **21** jegliche Aktivität. von vorbestimmter Priorität vollendet hat.

[0033] **Fig.** 4 zeigt eine beispielhafte Implementierung der Ausführungsform von **Fig.** 1 in einem Knoten eines zellularen Telekommunikationssystems; insbesondere in einer Basisstation (BTS) **40**. Die Basisstation (BTS) **40** enthält mehrere Übertrager/Empfänger-Karten (Tx/Rx) 42, welche über Luftschnittstelle **44** mit Mobilstationen, beispielsweise ein in **Fig.** 4 gezeigtes Benutzerequipment **46**, kommunizieren. Ebenfalls sind in Basisstation (BTS) **40** redundante Einheiten **40–21** und **40–22** enthalten, welche grundsätzlich erster CPU **21** und zweiter CPU **22** von der Ausführungsform von **Fig.** 1 entsprechen. Die Basisstation (BTS) **40** hat ebenfalls einen Erweiterungsanschluß (ET) 47 und einen Vermittler (switch) 48.

[0034] In der in **Fig.** 4 gezeigten bestimmten beispielhaften Basisstation (BTS) **40** ist der Vermittler **48** ein ATM (Asynchronous Transfer Mode) Vermittler, welcher ATM-Zellen zwischen Karten leitet, welche an Anschlüssen des Vermittlerkerns angebracht sind. Somit befinden sich in dem in **Fig.** 4 gezeigten Beispiel jede Übertrager/Empfänger-Karten (Tx/Rx) **42**, redundante Einheiten **40–21** und **40–22** und Erweiterungsanschlüsse (ET) **47** auf Schaltkarten, welche an entsprechenden Anschlüssen des Kerns von Vermittler **48** angeschlossen sind. Obwohl **Fig.** 4 die redundanten Einheiten **40–21** und **40–22** als direkt mit dem Vermittler **48** verbunden zeigt, sollte es verständlich sein, dass redundante Einheiten **40–21** und **40–22** sich vielmehr auf Karten befinden, welche an Vermittler **48** angeschlossen sind, und dass auf solchen Karten unterschiedliche andere Einrichtungen, beispielsweise Vermittleranschluß-Schnittstellenschaltungen, zwischen den redundanten Einheiten und Vermittler **48** eingesetzt werden können.

[0035] Der Erweiterungsanschluß (ET) **47** dient zur Verbindung von Basisstation (BTS) **40** über eine terrestrische Verbindung 49 an einen anderen Knoten des Telekommunikationssystems von **Fig.** 4, insbesondere an Funknetzwerk-Steuerer (RNC) **50**. In der Darstellung von **Fig.** 4 befindet sich der Resynchronisationsserver **50–24**, welcher Resynchronisationsserver **24** der Ausführungsform von **Fig.** 1 entspricht, im Funknetzwerk-Steuerer (RNC) **50**.

[0036] Der Aufbau und Betrieb von beispielhaften Knoten, beispielsweise Basisstation (BTS) **40** und Funknetzwerk-Steuerer (RNC) **50**, kann beispielsweise anhand der folgenden Patentanmeldungen verstanden werden: WO-A-99 33 314, veröffentlicht am 01.07.99, betreffend „Asynchronous Transfer Mode System"; WO-A-99 45 736, veröffentlicht am ,10.09.99, betreffend „Telecommunications Inter-Exchange Measurement Transfer"; WO-A-99 45 735, veröffentlicht am 10.09.99, betreffend „Telecommunications Inter-Exchange Congestion Control"; und WO-A-99 57 935, veröffentlicht am 11.11.99, betreffend „Inter-Exchange Paging".

[0037] In dem in **Fig.** 4 gezeigten bestimmten Beispiel können die redundanten Einheiten **40-21** und **40-22** unterschiedliche Aktivitäten für Basisstation (BTS) **40** durchführen, einschließlich unterschiedliche Steuerungsfunktionen. Eine solche Funktion ist das Durchführen von Funkverbindungsmessungen, beispielsweise Leistungsmessungen für jede Verbindung über Luftschnittstelle **44** und das Bestimmen eines Signal-zu-Rausch Abstandes für die Funkverbindung, mit Bezug zu jeder Verbindung. Eine solche Information ist wichtig in Mobilkommunikationen, beispielsweise CDMA-Mobilkommunikationen für Diversity und Soft Handover von einer Basisstation zu anderen Basisstationen. In eine solche Messung kann

tatsächlich eine der kritischen Operationen einbezogen sein, welche von einer der Zustandsmaschinen der redundanten Einheiten durchgeführt wird, welche vor Senden einer Resynchronisationsanforderung **2–3** gehandhabt werden muss. In der Ausführungsform von **Fig.** 4 handhabt jede der Zustandsmaschinen SM der CPU's einen Funkkanal oder eine ATM-Verbindung, oder steht damit in Verbindung.

[0038] **Fig.** 1B zeigt eine andere Ausführungsform der Erfindung, bei welcher sich Neustart-Logik **30B** in einem Synchronisationsserver **24** befindet. Der Resynchronisationsserver **24** kann entweder ein verschiedenartiger Knoten wie in **Fig.** 1 und **Fig.** 1B gezeigt sein, oder in einem der Einheiten-Knoten **21** oder **22** sein, wie in **Fig.** 1A gezeigt. **Fig.** 1C zeigt noch eine andere Ausführungsform der Erfindung, bei welcher sich Neustart-Logik **30C** in einem Knoten **60** befindet, welcher von einem Knoten der ersten Einheit **21**, dem Knoten von der zweiten Einheit **22** oder dem Knoten des Synchronisationsservers **24** verschieden ist. In beiden Ausführungsformen der **Fig.** 1B und **Fig.** 1C lernt die Neustart-Logik, dass eine CPU es beispielsweise bedarf neugebootet zu werden, und sendet Neustartmeldungen an jede Zustandsmaschine. Eine Darstellung des Sendens von solchen Neustartmeldungen wird durch Pfeil **1C–1** in **Fig.** 1C vorgesehen. Es sollte verstanden werden, dass die Neustartmeldungen in einem vorbestimmten Ablauf in der Weise von **Fig.** 3 oder gleichzeitig in der Weise von **Fig.** 3A gesendet werden können. Die Synchronisation von jeder Zustandsmaschine wird mit Bezug auf **Fig.** 2 verstanden, wie oben beschrieben.

[0039] Somit regt die vorliegende Erfindung die Zustandsmaschinen einer Reset-Einheit auf eine allmähliche, vorbestimmte Weise an, welches es jeder Zustandsmaschine erlaubt selber zu entscheiden, wann sie synchronisiert werden sollte. Darüber hinaus stellt die vorliegende Erfindung ein Schema für die. Zustandsmaschinen bereit, um einen dritten Beteiligten, beispielsweise den Server **24**, in dem Resynchronisationsprozeß zu gebrauchen. Natürlich muß die Reset-Einheit wissen, welche Zustandsmaschinen neu zu starten sind, und in welcher Reihenfolge oder Sequenz oder in welchem logischen Muster.

[0040] Wenn eine der CPU's (beispielweise eine Standby-CPU) es bedarf neu gestartet zu werden, wird somit das allmähliche Neustarten der vorliegenden Erfindung niemals die andere oder aktive CPU mit Bezug zur Außenwelt als verloren erscheinen lassen. Der einzige Teil des Gesamtsystems, welcher in Bezug zu seiner externen Umgebung als verloren erscheinen wird, ist eine beliebige Zustandsmaschine, welche sich derzeit selber resynchronisiert. Indem es jedoch jeder Zustandsmaschine überlassen wird zu entscheiden, wann eine Resynchronisierung durchzuführen ist, ist die Zeit, während der die Zustandsmaschine verloren erscheint, vernachlässigbar (da die Zustandsmaschine selber die für sie beste Zeit zur Resynchronisierung wissen oder voraussehen kann).

[0041] Es sollte verstanden werden, dass der Ausdruck „Einheit", wie hier verwendet, nicht notwendigerweise auf einen Prozessor beschränkt ist, welcher programmierte Instruktionen ausführt, sondern auf jeglichen Typ logischer Einrichtung oder Schaltung erweitert werden kann, welche mehrere Zustandsmaschinen überwacht. Auch darf die Tatsache, dass die hier dargestellten Zustandsmaschinen in Software realisiert werden, nicht ausschließen, dass. die Erfindung auf Hardware oder schaltungsgebundene Zustandsmaschinen angewendet werden kann, welche die Resynchronisationsprinzipien der vorliegenden Erfindung verwenden.

[0042] Auch sollte nicht eine bestimmte Tragweite der Tatsache beigelegt werden, dass die hier dargestellten Zustandsmaschinen so dargestellt sind, als dass sie nur eine Anregung empfangen. Eine solch vereinfachte Darstellung wurde, nur aus Gründen der Übersichtlichkeit bereitgestellt. Mehrere an die Zustandsmaschinen angelegten Anregungen werden natürlich durch die vorliegende Erfindung abgedeckt.

[0043] Der Ort von Resynchronisationsserver **24** und Resynchronisationsserver **24'**, wie hier beschrieben, ist nicht in dem Sinne beschränkt, so dass der Fachmann realisieren wird, dass die Resynchronisationsserver auf andere Weisen gesetzt und verbünden werden können, beispielsweise indem andere Elemente zwischen Resynchronisationsserver und einer oder mehreren der CPU's gesetzt werden kann.

[0044] Darüber hinaus ist die in **Fig.** 4 bereitgestellte, Darstellung von einer ATM basierten Basisstation (BTS) ein Beispiel dafür, wie die vorliegende Erfindung in einem Knoten eines Telekommunikationssystems implementiert werden kann. Der Knoten zur Implementierung braucht keine Basisstation zu sein, sondern kann ein anderer Knoten sein, wie beispielsweise der Funknetzwerk-Steuerer(RNC)-Knoten. Darüber hinaus. sollte verständlich sein, dass eine Verwendung der vorliegenden Erfindung natürlich nicht an eine Anwendung von ATM oder jeglichem anderen bestimmten Protokoll gebunden ist. Die vorliegende Erfindung kann zur Synchronisation von SSCOP-Verbindungen, Funkverbindungen und anderen Verbindungen in einem redundanten System verwendet werden.

[0045] Während die Erfindung in Verbindung mit dem beschrieben wurde, was derzeit als die praktischste und bevorzugteste Ausführungsform betrachtet wird, sollte es verstanden werden, dass die Erfindung nicht auf die beschriebene Ausführungsform beschränkt ist, sondern im Gegenteil dazu dient, unterschiedliche Modifikationen und äquivalente Anordnungen abzudecken, welche innerhalb des Umfangs der anhängenden Ansprüche enthalten sind.

ZUSAMMENFASSUNG

[0046] Ein redundantes System (**20**) enthält eine

erste Einheit (**21**), eine. zweite Einheit (**22**) und einen Synchronisationsserver (**24**). Die erste Einheit (**21**) und eine zweite Einheit (**22**) enthalten jeweils mehrere Zustandsmaschinen (SMs) zur Durchführung von Aufgaben. Zumindest einige, der Zustandsmaschinen, welche an der ersten Einheit bereitgestellt sind, führen gleichzeitig selbe Aufgaben durch, wie zumindest entsprechende von einigen der Zustandsmaschinen, welche an der zweiten Einheit bereitgestellt sind. Wenn eine Zustandsmaschine der zweiten Einheit es bedarf resynchronisiert zu werden, empfängt der Synchronisationsserver (**24**) eine Resynchronisationsanforderung (**2–1**) von der Resynchronisation anfordernden Zustandsmaschine der zweiten Einheit, und stellt darauf eine out-of-synchronization Anzeige (**2–2**) an einer entsprechenden Zustandsmaschine der ersten Einheit bereit. In Ansprechen auf die out-of-synchronization Anzeige (**2–2**) von dem Synchronisationsserver erzeugt die entsprechende Zustandsmaschine der ersten Einheit eine Resynchronisationsanforderung (**2–3**), zu einer Zeit, welche von der entsprechenden Zustandsmaschine in der ersten Einheit als geeignet erachtet wird. In Ansprechen auf die Resynchronisationsanforderung (**2–3**) von der entsprechenden Zustandsmaschine gibt der Synchronisationsserver einen Resynchronisationsbefehl (**2–4**) sowohl an die Resynchronisation anfordernde Zustandsmaschine der zweiten Einheit, als auch an die entsprechende Zustandsmaschine der ersten Einheit aus, wodurch simultan eine Resynchronisation verursacht wird. Zumindest eine der ersten Einheit und der zweiten Einheit enthält eine Neustart-Logik, welche eine Neustart-Reihenfolge der Mehrzahl von Zustandsmaschinen vorschreibt, wodurch eine graduelle Anregung der Zustandsmaschinen der redundanten Einheiten bereitgestellt wird. Danach steuert jede der Mehrzahl von Zustandsmaschinen ihre eigene Resynchronisat on.

Patentansprüche

1. Ein redundantes System mit:
einer ersten Einheit und einer zweiten Einheit, welche jeweils mehrere Zustandsmaschinen enthalten, zum Durchführen von Aufgaben, wobei zumindest einige der Zustandsmaschinen, welche an der ersten Einheit bereitgestellt sind, gleichzeitig dieselben Aufgaben durchführen, wie zumindest entsprechende von einigen der Zustandsmaschinen, welche an der zweiten Einheit bereitgestellt sind;
einem Synchronisationsserver, welcher eine Resynchronisationsanforderung von einer Resynchronisationsanforderung-Zustandsmaschine von der zweiten Einheit empfängt und darauf eine Ausser-Synchronisation-Anzeige an eine entsprechende Zustandsmaschine von der ersten Einheit bereitstellt;
wobei in Ansprechen auf die Ausser-Synchronisation-Anzeige von dem Synchronisationsserver, die entsprechende Zustandsmaschine von der ersten Einheit eine Resynchronisationsanforderung erzeugt;
wobei in Ansprechen auf die Resynchronisationsanforderung von der Zustandsmaschine von der ersten Einheit, der Synchronisationsserver einen Resynchronisationsbefehl ausgibt, sowohl an die entsprechende Zustandsmaschine von der ersten Einheit, als auch an die Resynchronisationsanforderung-Zustandsmaschine von der zweiten Einheit;
wobei die Resynchronisationsanforderung-Zustandsmaschine von der zweiten Einheit und die entsprechende Zustandsmaschine von der ersten Einheit gleichzeitig resynchronisieren, in Ansprechen auf den Resynchronisationsbefehl.

2. Vorrichtung nach Anspruch 1, wobei die entsprechende Zustandsmaschine von der ersten Einheit eine Resynchronisationsanforderung nur dann erzeugt, nachdem die entsprechende Zustandsmaschine von der ersten Einheit eine Aktivität von vorbestimmter Priorität vollendet hat.

3. Vorrichtung nach Anspruch 1, wobei mindestens eine von der ersten Einheit und der zweiten Einheit eine Neustart-Logik enthält, welche eine Neustart-Reihenfolge von den mehreren Zustandsmaschinen vorschreibt.

4. Vorrichtung nach Anspruch 3, wobei die Neustart-Logik einen Neustart von den mehreren Zustandsmaschinen auf einer Basis von einer Zustandsmaschine zu einer Zeit vorschreibt.

5. Vorrichtung nach Anspruch 1, wobei der Synchronisationsserver und/oder eine dritte Einheit eine Neustart-Logik enthält, welche eine Neustart-Reihenfolge der mehreren Zustandsmaschinen vorschreibt.

6. Vorrichtung nach Anspruch 5, wobei die Neustart-Logik einen Neustart von den mehreren Zustandsmaschinen auf einer Basis von einer Zustandsmaschine zu einer Zeit vorschreibt.

7. Vorrichtung nach Anspruch 1, wobei eine Neustart-Logik mehrere Zustandsmaschinen von einer Einheit neustartet, und wobei nach Neustart jede Zustandsmaschine von der Einheit entscheidet, wann sie resynchronisieren wird.

8. Vorrichtung nach Anspruch 1, wobei die erste Einheit und die zweite Einheit sich in einem Knoten von einem zellularen Telekommunikationssystem befinden.

9. Vorrichtung nach Anspruch 1, wobei der Synchronisationsserver in einer von der ersten Einheit und der zweiten Einheit enthalten ist.

10. Vorrichtung nach Anspruch 1, wobei der Synchronisationsserver sich an einem Knoten befindet, welcher nicht die erste Einheit oder die zweite Einheit

enthält.

11. Vorrichtung nach Anspruch 1, wobei mindestens einige Zustandsmaschinen an der ersten Einheit und der zweiten Einheit synchronisiert werden durch externe Anregung.

12. Verfahren zum Betreiben eines redundanten Systems, welches eine erste Einheit und eine zweite Einheit hat, wobei jede von der ersten Einheit und der zweiten Einheit mehrere Zustandsmaschinen zum Durchführen von Aufgaben enthält, wobei mindestens einige der Zustandsmaschinen, welche an der ersten Einheit bereitgestellt werden, gleichzeitig dieselben Aufgaben durchführen, wie zumindest entsprechende von einigender Zustandsmaschinen, welche an der zweiten Einheit bereitgestellt werden; wobei das Verfahren die folgenden Schritte enthält:
(1) Empfangen einer Resynchronisationsanforderung an einem Synchronisationsserver von einer Resynchronisationsanforderung-Zustandsmaschine von der zweiten Einheit;
(2) in Ansprechen auf Schritt (1), Bereitstellen einer Ausser-Synchronisation-Anzeige an einer entsprechenden Zustandsmaschine von der ersten Einheit;
(3) in Ansprechen auf Schritt (2), Erzeugen an der entsprechenden Zus andsmaschine von der ersten Einheit einer entsprechenden Zustandsmaschinen Resynchronisationsanforderung;
(4) in Ansprechen auf Schritt (3), Ausgeben eines Resynchronisationsbefehls von dem Synchronisationsserver, sowohl an die Resynchronisationsanforderung-Zustandsmaschine von der zweiten Einheit und der ersten Einheit, als auch an die entsprechende Zustandsmaschine von der ersten Einheit;
(5) gleichzeitiges Resynchronisieren der Resynchronisationsanforderung-Zustandsmaschine von der zweiten Einheit und der entsprechenden Zustandsmaschine von der ersten Einheit, in Ansprechen auf den Resynchronisationsbefehl.

13. Verfahren nach Anspruch 12, ferner enthaltend ein Durchführen von Schritt (3), nur nachdem die entsprechende Zustandsmaschine von der ersten Einheit eine Aktivität von vorbestimmter Priorität vollendet hat.

14. Verfahren nach Anspruch 12, ferner enthaltend einen Neustart der mehreren Zustandsmaschinen von mindestens einer von der ersten Einheit und der zweiten Einheit, in Übereinstimmung mit einem vorbestimmten Befehl.

15. Verfahren nach Anspruch 14, ferner enthaltend einen Neustart der mehreren Zustandsmaschinen von mindestens einer von der ersten Einheit und der zweiten Einheit, bei einer Rate von einer Zustandsmaschine zu einer Zeit.

16. Verfahren nach Anspruch 13, ferner enthaltend eine Neustart-Logik, welche mehrere Zustandsmaschinen von einer Einheit neustartet, und wobei nach Neustart jede Zustandsmaschine von der Einheit entscheidet, wann sie resynchronisieren wird.

17. Verfahren nach Anspruch 13, ferner enthaltend eine Unterbringung von der ersten Einheit und der zweiten Einheit in einem Knoten von einem zellularen Telekommunikationssystem.

18. Verfahren nach Anspruch 13, ferner enthaltend ein Aufnehmen des Synchronisationsservers in einer von der ersten Einheit und der zweiten Einheit.

19. Verfahren nach Anspruch 13, ferner enthaltend ein Aufnehmen des Synchronisationsservers in den Synchronisationsserver oder einer dritten Einheit.

20. Verfahren nach Anspruch 13, ferner enthaltend ein Verwenden externer Anregung, um mindestens einige Zustandsmaschinen an der ersten Einheit und der zweiten Einheit während eines gewöhnlichen Betriebes zu synchronisieren.

21. Telekommunikationssystem mit:
einem Knoten, welcher ein redundantes System hat, welches eine erste Einheit, und eine zweite Einheit hat, wobei jede mehrere Zustandsmaschinen enthält, um Aufgaben von dem Knoten durchzuführen, wobei mindestens einige der Zustandsmaschinen, welche an der ersten Einheit bereitgestellt sind, gleichzeitig dieselben Aufgaben durchführen, wie mindestens entsprechende von einigen der Zustandsmaschinen, welche an der zweiten Einheit bereitgestellt sind;
einem Synchronisationsserver, welcher eine Resynchronisationsanforderung empfängt von einer Resynchronisationsanforderung-Zustandsmaschine von der zweiten Einheit, und darauf eine Ausser-Synchronisation-Anzeige bereitstellt an einer entsprechenden Zustandsmaschine von der ersten Einheit;
wobei in Ansprechen auf die Ausser-Synchronisation-Anzeige von dem Synchronisationsserver, die entsprechende Zustandsmaschine von der ersten Einheit eine Resynchronisationsanforderung erzeugt;
wobei in Ansprechen auf die Resynchronisationsanforderung von der entsprechenden Zustandsmaschine von der ersten Einheit, der Synchronisationsserver einen Resynchronisationsbefehl ausgibt, sowohl an die Resynchronisationsanforderung-Zustandsmaschine von der zweiten Einheit, als auch an die entsprechende Zustandsmaschine von der ersten Einheit;
wobei die Resynchronisationsanforderung-Zustandsmaschine von der zweiten Einheit und die entsprechende Zustandsmaschine von der ersten Einheit gleichzeitig resynchronisieren, in Ansprechen auf den Resynchronisationsbefehl.

22. Vorrichtung nach Anspruch 21, wobei die entsprechende Zustandsmaschine von der ersten Einheit eine Resynchronisationsanforderung nur dann erzeugt, nachdem die entsprechende Zustandsmaschine von der ersten Einheit eine Aktivität von vorbestimmter Priorität vollendet hat.

23. Vorrichtung nach Anspruch 21, wobei mindestens eine der ersten Einheit und der zweiten Einheit eine Neustart-Logik enthält, welche eine Neustart-Reihenfolge von den mehreren Zustandsmaschinen vorschreibt.

24. Vorrichtung nach Anspruch 21, wobei die Neustart-Logik einen Neustart von den mehreren Zustandsmaschinen vorschreibt, auf einer Basis von einer Zustandsmaschine zu einer Zeit.

25. Vorrichtung nach Anspruch 21, wobei eine Neustart-Logik mehrere Zustandsmaschinen von einer Einheit neustartet, und wobei nach Neustart jede Zustandsmaschine von der Einheit entscheidet, wann sie resynchronisieren wird.

26. Vorrichtung nach Anspruch 21, wobei der Synchronisationsserver in einer von der ersten Einheit und der zweiten Einheit enthalten ist.

27. Vorrichtung nach Anspruch 21, wobei mindestens einige Zustandsmaschinen an der ersten Einheit und der zweiten. Einheit durch externe Anregung synchronisiert werden.

28. Verfahren zum Betreiben eines Telekommunikationssystems, welches einen Knoten umfaßt, welcher ein redundantes System hat, welches eine erste Einheit und eine zweite Einheit enthält, wobei jede von der ersten Einheit und der zweiten Einheit mehrere Zustandsmaschinen zum Durchführen von Aufgaben von dem Knoten enthält, wobei mindestens einige der Zustandsmaschinen, welche an der ersten Einheit bereitgestellt werden, gleichzeitig dieselben Aufgaben durchführen, wie zumindest entsprechende von einigen der Zustandsmaschinen, welche an der zweiten Einheit bereitgestellt werden; wobei das Verfahren die folgenden Schritte enthält:
(1) Empfangen einer Resynchronisationsanforderung an einem Synchronisationsserver von einer Resynchronisationsanforderung-Zustandsmaschine von der zweiten Einheit;
(2) in Ansprechen auf Schritt (1), Bereitstellen einer Ausser-Synchronisation-Anzeige an einer entsprechenden Zustandsmaschine von der ersten Einheit;
(3) in Ansprechen auf Schritt (2), Erzeugen an der entsprechenden Zustandsmaschine von der ersten Einheit einer entsprechenden Zustandsmaschinen Resynchronisationsanforderung;
(4) in Ansprechen auf Schritt (3), Ausgeben eines Resychronisationsbefehls von dem Synchronisationsserver, sowohl an die Resynchronisationsanforderung-Zustandsmaschine von der zweiten Einheit und der ersten Einheit, als auch an die entsprechende Zustandsmaschine von der ersten Einheit;
(5) gleichzeitiges Resynchron sieren der Resynchronisationsanforderung-Zustandsmaschine von der zweiten Einheit und der entsprechenden Zustandsmaschine von der ersten Einheit, in Ansprechen auf den Resynchron sationsbefehl,

29. Verfahren nach Anspruch 28, ferner enthaltend ein Durchführen von Schritt (3), nur nachdem die entsprechende Zustandsmaschine von der ersten Einheit eine Aktivität von vorbestimmter Priorität vollendet hat.

30. Verfahren nach Anspruch 28, ferner enthaltend einen Neustart der mehreren Zustandsmaschinen von mindestens einer von der ersten Einheit und der zweiten Einheit, in Übereinstimmung mit einem vorbestimmten Befehl.

31. Verfahren nach Anspruch 30, ferner enthaltend einen Neustart der mehreren Zustandsmaschinen von mindestens einer von der ersten Einheit und der zweiten Einheit, bei einer Rate von einer Zustandsmaschine zu einer Zeit.

32. Verfahren nach Anspruch 28, ferner enthaltend eine Neustart-Logik, welche mehrere Zustandsmaschinen von einer Einheit neustartet, und wobei nach Neustart jede Zustandsmaschine von der Einheit entscheidet, wann sie resynchronisieren wird.

33. Verfahren nach Anspruch 28, ferner enthaltend ein Aufnehmen des Synchronisationsservers in einer von der ersten Einheit und der zweiten Einheit.

34. Verfahren nach Anspruch 28, ferner enthaltend ein Verwenden externer Anregung; um mindestens einige Zustandsmaschinen an der ersten Einheit und der zweiten Einheit während eines gewöhnlichen Betriebes zu synchronisieren.

Es folgen 8 Blatt Zeichnungen

Anhängende Zeichnungen

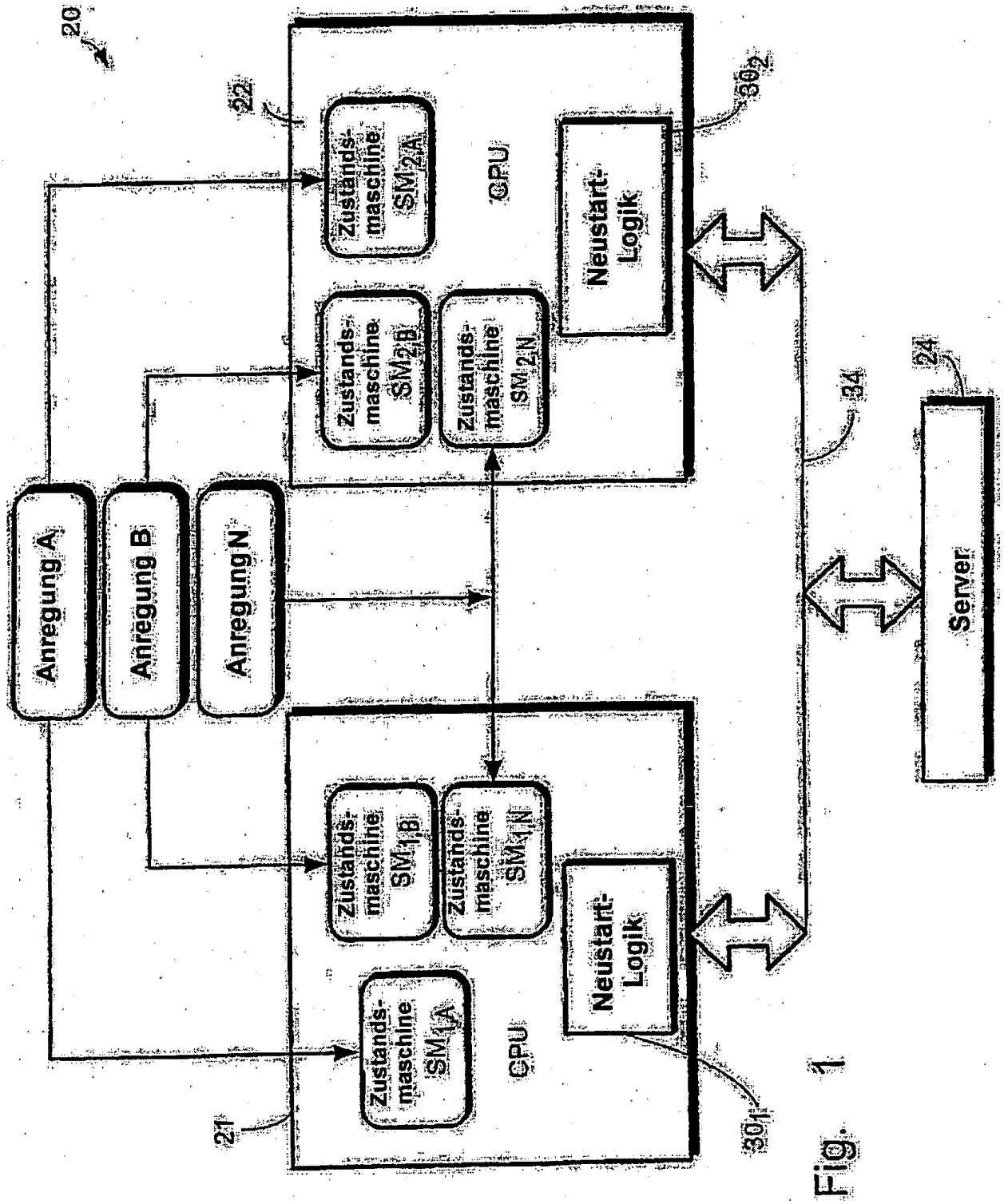


Fig. 1

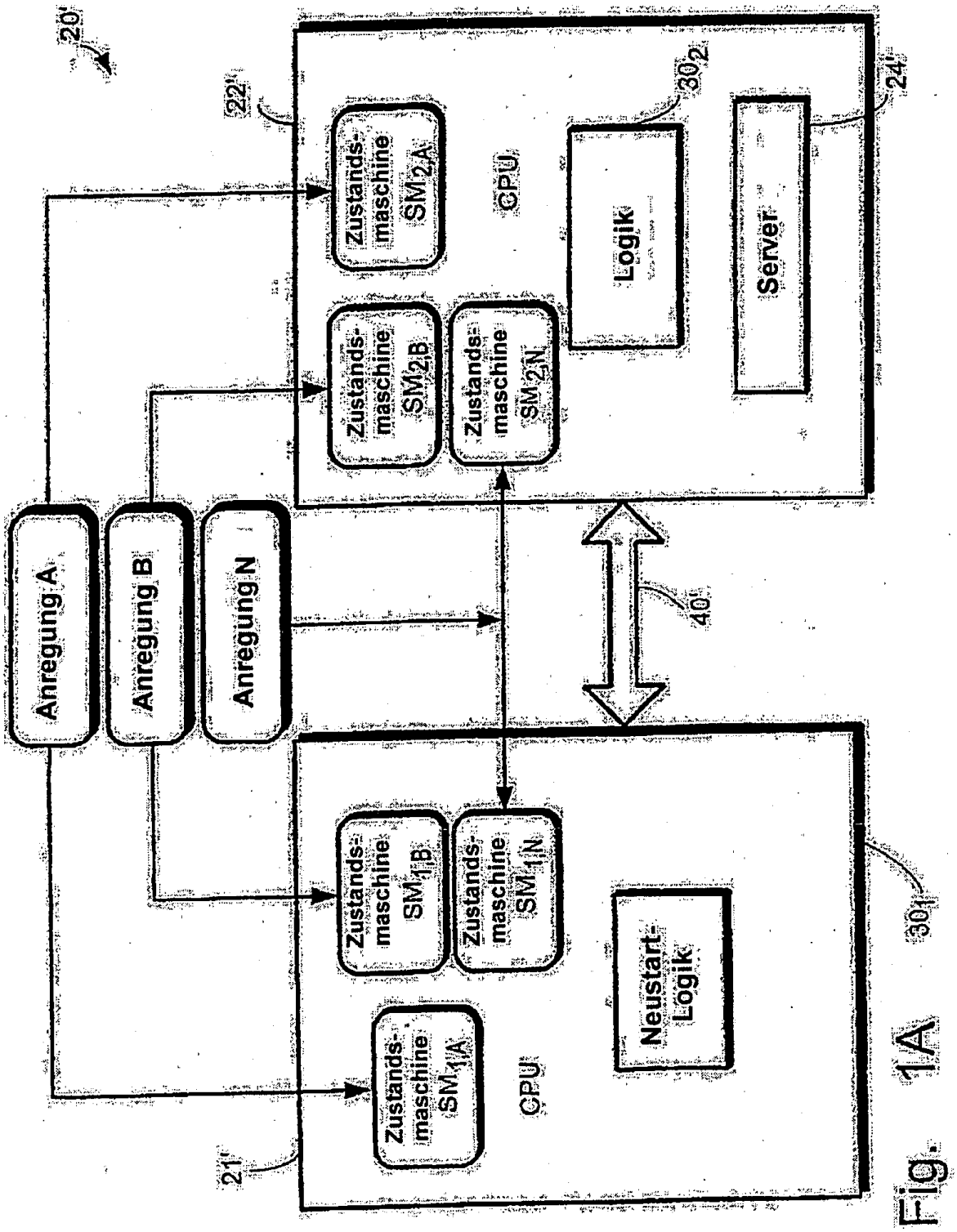

20'
22'
24'
30_2
Zustands-maschine SM_{2A}
CPU
Logik
Server
Zustands-maschine SM_{2B}
Zustands-maschine SM_{2N}
Anregung A
Anregung B
Anregung N
40'
Zustands-maschine SM_{1B}
Zustands-maschine SM_{1N}
Neustart-Logik
30_1
Zustands-maschine SM_{1A}
CPU
21'

Fig. 1A

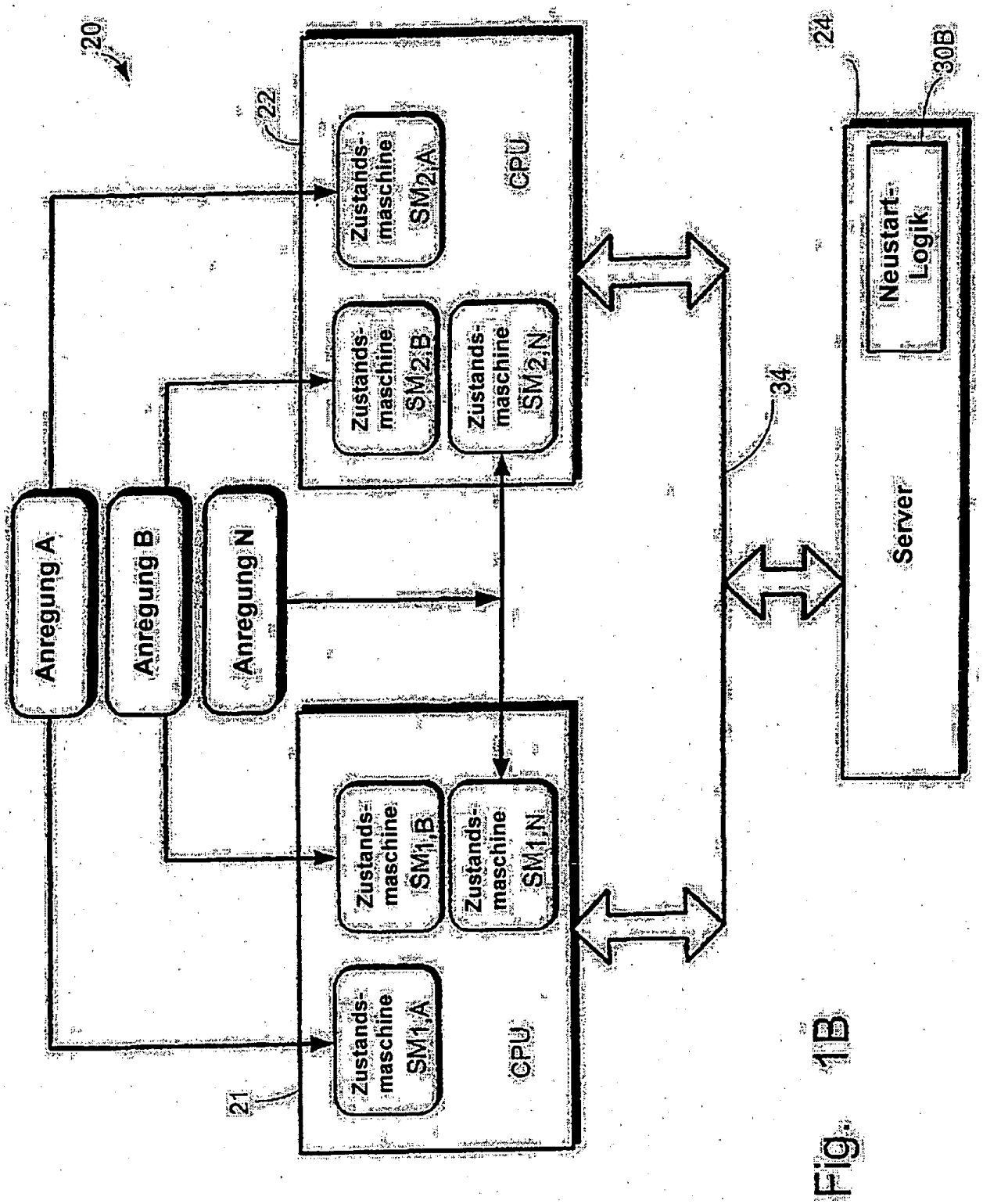

20
22
24
30B
Zustands-maschine SM2,A
CPU
Zustands-maschine SM2,B
Zustands-maschine SM2,N
Neustart-Logik
34
Anregung A
Anregung B
Anregung N
Server
Zustands-maschine SM1,B
Zustands-maschine SM1,N
Zustands-maschine SM1,A
CPU
21

Fig. 1B

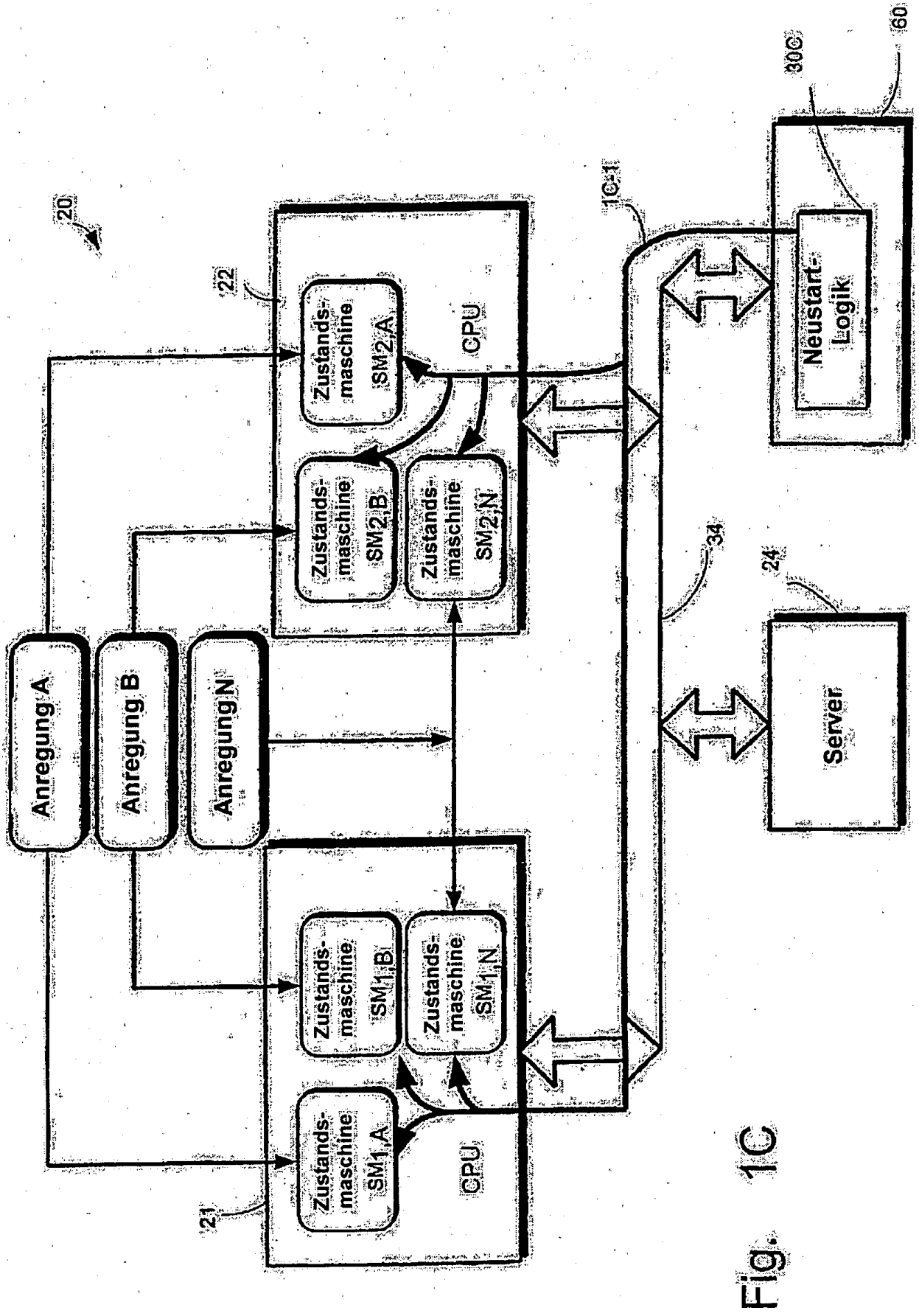

20
22
21
CPU
CPU
Zustands-maschine SM2,A
Zustands-maschine SM2,B
Zustands-maschine SM2,N
Zustands-maschine SM1,A
Zustands-maschine SM1,B
Zustands-maschine SM1,N
Anregung A
Anregung B
Anregung N
1C-1
30C
60
Neustart-Logik
34
24
Server

Fig. 1C

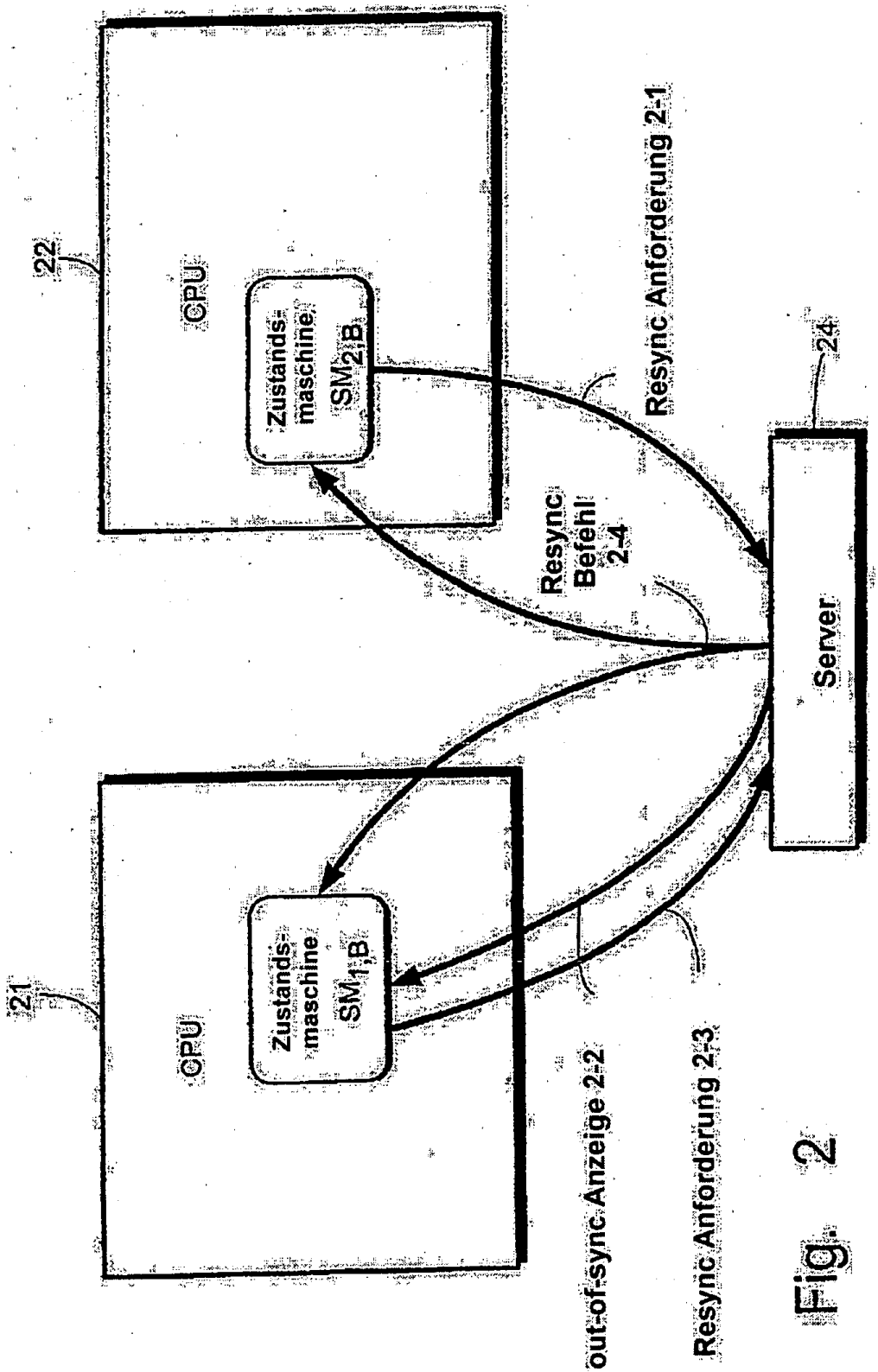

22
CPU
Zustands-maschine SM2B
Resync Anforderung 2-1
24
Server
Resync Befehl 2-4
21
CPU
Zustands-maschine SM1B
out-of-sync Anzeige 2-2
Resync Anforderung 2-3

Fig. 2

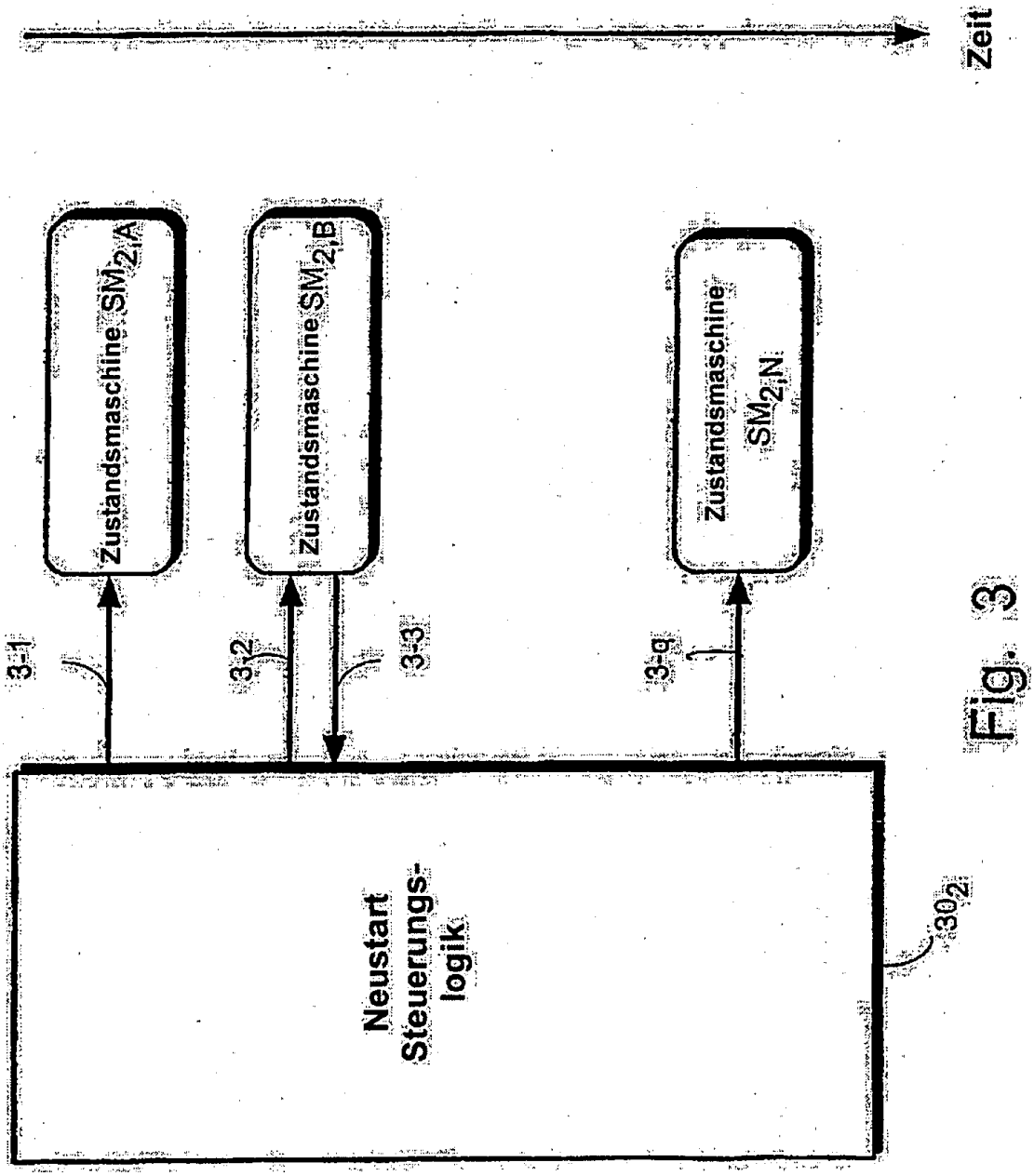

Zeit
Zustandsmaschine SM_2,A
Zustandsmaschine SM_2,B
Zustandsmaschine SM_2,N
3-1
3-2
3-3
3-q
Neustart Steuerungs-logik
30_2

Fig. 3

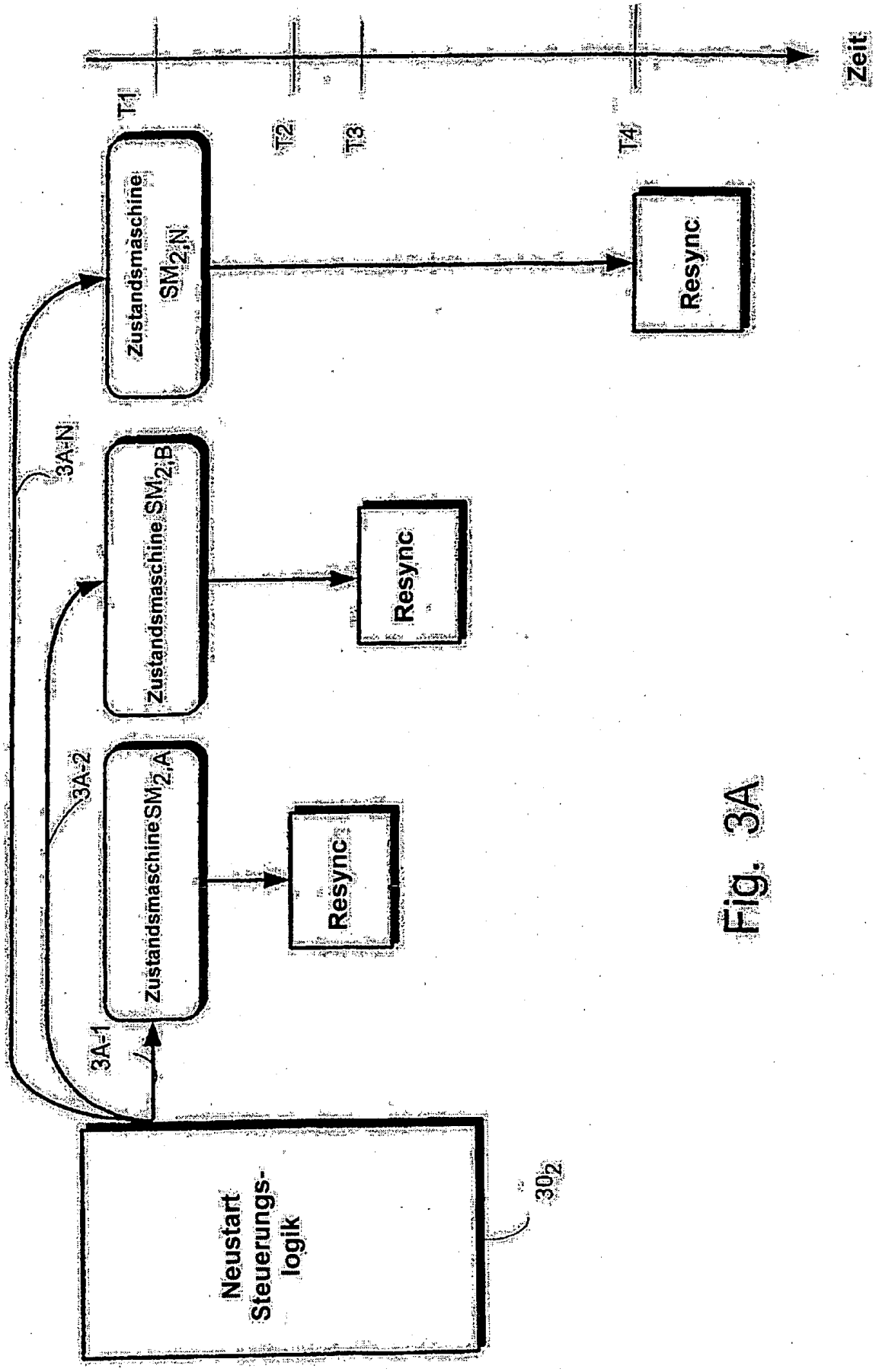

Zeit
T1
T2
T3
T4
Zustandsmaschine SM2,N
Resync
3A-N
Zustandsmaschine SM2,B
Resync
3A-2
Zustandsmaschine SM2,A
Resync
3A-1
Neustart Steuerungs-logik
302

Fig. 3A

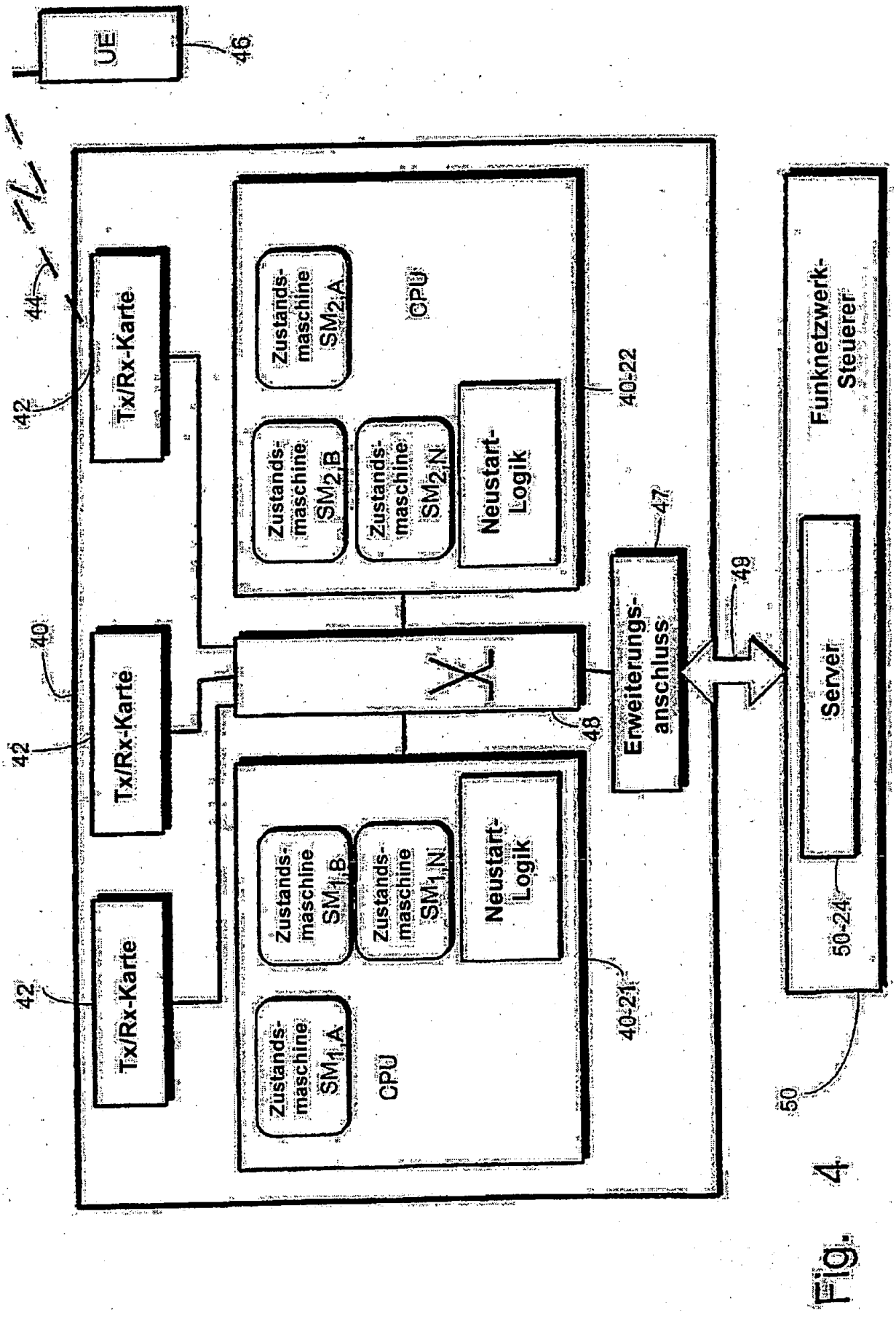

UE
46
44
42
Tx/Rx-Karte
Tx/Rx-Karte
Tx/Rx-Karte
40
42
42
42
Zustands-maschine SM2,A
Zustands-maschine SM2,B
Zustands-maschine SM2,N
Neustart-Logik
CPU
40-22
Zustands-maschine SM1,A
Zustands-maschine SM1,B
Zustands-maschine SM1,N
Neustart-Logik
CPU
40-21
48
47
Erweiterungs-anschluss
49
Server
50-24
Funknetzwerk-Steuerer
50

Fig. 4