



(19)  
Bundesrepublik Deutschland  
Deutsches Patent- und Markenamt

(10) DE 697 32 274 T2 2005.07.28

(12)

## Übersetzung der europäischen Patentschrift

(97) EP 0 823 166 B1

(21) Deutsches Aktenzeichen: 697 32 274.2

(86) PCT-Aktenzeichen: PCT/US97/02664

(96) Europäisches Aktenzeichen: 97 907 725.2

(87) PCT-Veröffentlichungs-Nr.: WO 97/031459

(86) PCT-Anmeldetag: 21.02.1997

(87) Veröffentlichungstag

der PCT-Anmeldung: 28.08.1997

(97) Erstveröffentlichung durch das EPA: 11.02.1998

(97) Veröffentlichungstag

der Patenterteilung beim EPA: 19.01.2005

(47) Veröffentlichungstag im Patentblatt: 28.07.2005

(51) Int Cl.<sup>7</sup>: H04L 12/56

H04Q 11/04, G06F 13/38

(30) Unionspriorität:

603913 22.02.1996 US

(84) Benannte Vertragsstaaten:

DE, FR, GB

(73) Patentinhaber:

Fujitsu Ltd., Kawasaki, Kanagawa, JP

(72) Erfinder:

WICKI, M., Thomas, Palo Alto, US; HELLAND, J.,  
Patrick, Redmond, US; LARSON, D., Jeffrey, San  
Jose, US; MU, Albert, Milpitas, US; SASTRY,  
Raghu, Santa Clara, US; SCHOBER, L., Richard,  
Cupertino, US

(74) Vertreter:

W. Seeger und Kollegen, 81369 München

(54) Bezeichnung: FLUSSSTEUERUNGSPROTOKOLLSYSTEM UND VERFAHREN

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 eingelebt, 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****HINTERGRUND DER ERFINDUNG****1. Gebiet der Erfindung**

**[0001]** Die vorliegende Erfindung betrifft im allgemeinen das Gebiet des Sendens von Daten im Netz und im besonderen das Gebiet von Sendeprotokollsystemen und -verfahren im verteilten Netz.

**2. Beschreibung der Hintergrundtechnik**

**[0002]** Es gibt verschiedene Techniken zum effektiven und zuverlässigen Übertragen von Informationen zwischen Terminals (Knoten) eines Computernetzes. Eine solche Technik ist die Paketvermittlung. Bei der Paketvermittlung sendet ein Sende-(Quellen)-Knoten eine Meldung an einen Empfangs-(Ziel)-Knoten. Die Meldung wird in viele Abschnitte mit variabler Größe geteilt. Diese Abschnitte werden als Pakete bezeichnet. Jedes Paket enthält einen Datenabschnitt, einen Paketkopf und häufig Fehlerdetektionsinformationen, wie z. B. Paritätsinformationen. Der Datenabschnitt enthält die Informationen in der Meldung, die zu senden ist, zusammen mit anderen Protokollinformationen von höheren Protokollsichten in dem Netz wie z. B. der Anwendungsschicht, der Darstellungsenschicht, der Kommunikationssteuerungsschicht, so wie es in dem ISO-Referenzmodell angegeben ist. Der Paketkopf enthält Informationen, die sich, unter anderen Informationen, auf den Ort des Pakets in der Paketfolge beziehen.

**[0003]** Um die Pakete zwischen Knoten des Netzes zu transportieren, sind zusätzliche Informationen erforderlich. Diese zusätzlichen Informationen werden oft in einem Rahmenkopf gespeichert. Ein Rahmenkopf wird zu den Paketen hinzugefügt, und die Kombination aus einem Paket und einem Rahmenkopf wird als Rahmen bezeichnet. Jedes Netz begrenzt die Größe eines Rahmens, und deshalb wird, falls die Meldung zu groß ist, um in einen einzelnen Rahmen zu passen, die Meldung in zwei oder mehr Rahmen getrennt. Die Informationen, die in dem Rahmenkopf enthalten sind, enthalten Informationen, die den endgültigen Zielknoten für den Rahmen identifizieren.

**[0004]** Rahmen werden durch das Netz unter Verwendung von Routern gesendet. Zum Beispiel kann ein Rahmen von einem ersten Router zu einem zweiten Router, dann von dem zweiten Router zu einem dritten Router und dann von dem dritten Router zu dem Zielknoten gesendet werden. Falls die Rate des ankommenden Verkehrs an einem ersten Router größer als die maximale Rahmensederate des ersten Routers ist, wird sich dann ein Rahmenpuffer in dem ersten Router füllen, und schließlich wird entweder der zweite Router den ersten Router instruieren, das Senden von Daten temporär zu stoppen, oder emp-

fangene Rahmen werden Rahmen überschreiben, die zuvor in dem Puffer gespeichert wurden, und diese überschriebenen Rahmen gehen verloren. Die meisten Systeme verwenden ein Sendesteuersystem, bei dem der Empfangsrouter, z. B. der zweite Router, ein Steuersignal zu dem Senderouter senden kann, das den Senderouter instruiert, das Senden temporär zu stoppen.

**[0005]** Herkömmliche Sendesteuersysteme nutzen entweder eine einzelne Signalleitung sowohl für Datenrahmen als auch für Steuersignale oder eine dedizierte Steuersignalleitung. Ein Beispiel für ein herkömmliches Sendesteuersystem ist ein RS232-Sendeprotokoll. Bei dem RS232-Sendeprotokoll sendet ein Sendenetzelement, z. B. ein Router, Rahmen über eine Rahmensignalleitung zu einem Empfangsnetzelement. Falls der Empfangsrouter das Senden von Rahmen durch den Senderouter stoppen möchte, erzeugt der Empfangsrouter ein Sende-Aus-(XOFF)-Steuersignal und sendet das Steuersignal über die Rahmensignalleitung zu dem Senderouter.

**[0006]** In Hochfrequenznetzen mit z. B. 500 MHz wird durch die Zeit, die erforderlich ist, bis das Steuersignal den Senderouter erreicht, eine signifikante Latenz in das System eingeschleppt. Genauer gesagt, falls eine Steuersignalleitung verwendet wird, um ein Steuersignal von dem Empfangsrouter zu dem Senderouter zu transportieren, ist dann die Latenz, die eingeschleppt wird, ungefähr der Zeit gleich, die ein Signal benötigt, um die Streckendistanz zwischen dem Senderouter und dem Empfangsrouter zurückzulegen. Falls die Streckendistanz ungefähr zwei Meter (2 m) beträgt und die Taktfrequenz 500 MHz ist, wobei bei jedem Taktzyklus ein Datenbit auf jeder Signalleitung auf der Strecke gesendet wird, werden dann ungefähr 3,3 Bits (3,3 Bits = 500 MHz/300 km/s × 2 m) auf jeder Signalleitung durch den Senderouter zwischen der Zeit, wenn der Empfangsrouter das Steuersignal sendet, und der Zeit, wenn der Senderouter das Signal empfängt, gesendet. Zusätzlich ist laufend dieselbe Anzahl von Bits auf der Signalleitung, weshalb durch den Empfangsrouter nach dem Senden des XOFF-Steuersignals ungefähr 6,6 Bits empfangen werden. Falls viele Signalleitungen auf der Strecke vorhanden sind, z. B. 32, können dann ungefähr 212 Bits (6,6 Bits × 32 Signalleitungen) durch dem Empfangsrouter empfangen werden, nachdem der Senderouter instruiert wurde, das Senden zu stoppen. Diese 212 Bits müssen in einem Puffer in dem Empfangsrouter gespeichert werden, oder die Daten gehen sonst verloren. Um diese Bits zu speichern, muß der Empfangsrouter Extrapuffer unterhalten, um all diese Bits zu erfassen. Puffer sind teuer, und eine ineffektive Nutzung von Puffern kann einen signifikanten Einfluß auf die Leistung eines Netzes haben. Daher bedeutet das Einrichten von ausreichendem Pufferraum, um wenigstens 212

Bits zu speichern, eine ineffektive Nutzung von Systembetriebsmitteln. Bei dem vorliegenden Beispiel werden 212 Pufferräume reserviert, und wenn die übrigen Pufferräume in dem Empfangsrouter voll sind, sendet der Empfangsrouter ein XOFF-Steuersignal, um zu gewährleisten, daß alle Daten, die durch den Datensenderrouter zwischen dem Senden des XOFF-Steuersignals durch den Datenempfangsrouter und dem Empfang des XOFF-Steuersignals durch den Datensenderrouter gesendet werden, erfaßt und in einem Pufferraum gespeichert werden. Jedoch sind nicht alle von diesen Bits unbedingt Teil eines Datenrahmens. Oft sind manche der Bits "Leerlauf"-Bits oder andere "Abfall"-Bits, die über das Netz gesendet werden, wenn gerade keine Datenrahmen oder Steuersignale gesendet werden. Daher ist es nicht notwendig, diese Leerlaufbits zu speichern. Wenn Leerlaufbits gesendet werden, werden deshalb nicht alle von den reservierten Pufferräumen verwendet. Puffer sind teuer, wie oben beschrieben, und das RS232-Sendeprotokoll stellt eine ineffektive Nutzung dieser Puffer dar.

**[0007]** Wenn das RS232-Sendeprotokoll und eine einzelne Steuersignalleitung verwendet werden, beendet der Senderouter das Senden von Daten nach dem Empfang des XOFF-Signals. Wenn ausreichende Puffer in dem Empfangsrouter zur Verfügung stehen, erzeugt der Empfangsrouter ein Weiterversende-(XON)-Signal und sendet das XON-Signal zu dem Senderouter. Nach dem Empfang des XON-Signals wird der Senderouter das Senden von Daten wieder aufnehmen. Dieses System schlept jedoch eine Latenz in das Netz ein, die der Zeit zwischen dem Senden des XON-Steuersignals und dem Empfang der ersten Datenverbindung gleich ist.

**[0008]** Falls das Netz anstelle der Verwendung einer separaten Steuersignalleitung eine einzelne Datenleitung sowohl für Steuersignale, die durch den Empfänger gesendet werden, als auch für Datenrahmen, die durch den Sender gesendet werden, verwendet, nimmt die Latenz des Systems dann weiter zu, da der Senderouter das Senden stoppen muß, damit der Empfangsrouter ein Steuersignal senden kann. Daher wird eine Latenzperiode, die der Zeit ungefähr gleich ist, die ein Signal benötigt, um von dem Senderouter zu dem Empfangsrouter und zurück zu gelangen, in das Netz eingeschleppt.

**[0009]** EP-A-0 378 401 offenbart ein Datenkommunikationssystem, das einen Sender umfaßt, der ein Mittel zum Senden von Blöcken von Daten und zum Empfangen von Bestätigungsmeldungen enthält, eine Übertragungsleitung, die mit dem Sender gekoppelt ist, zur Kommunikation von Daten und Meldungen, und einen Empfänger, der mit der Übertragungsleitung gekoppelt ist. Gemäß jenem bekannten System zählt der Sender Bestätigungsmeldungen, die von dem Empfänger empfangen werden, um zu

bestimmen, ob ein Puffer zum Datenempfang bereit ist.

**[0010]** EP-A-0 380 856 beschreibt ein Verfahren in einer Vorrichtung zum Anschließen einer Systemsteuereinheit für einen Multiprozessor. Gemäß dem bekannten Verfahren werden Datenpakete in der Warteschlange zwischen der Systemsteuereinheit eines Multiprozessor-Computersystems zu ihren zugeordneten Eingabe/Ausgabe-(I/O)-Vorrichtungen jedes Systems übertragen. Die Senderarchitektur enthält einen Senderdatenpuffer, der Daten temporär speichert, um zu einem Empfängerpuffer gesendet zu werden, der eine Größe hat, die dem Sender bekannt ist. Jede gesendete Aufforderung enthält einen Zeiger auf ein Datenpaket in dem Sendepuffer und wird in der Warteschlange eingereiht. Der Sender verwendet jene Warteschlange, um zu identifizieren, welche Daten als nächste zu senden sind. Die Warteschlange gibt jedoch nicht die Verfügbarkeit eines Empfangspuffers zum Empfangen der Daten an.

**[0011]** US 5,473,755 ist auf ein System zum Steuern eines Datenstroms zwischen elektronischen Komponenten gerichtet. Die Empfangskomponente jenes Systems inkorporiert ein Mittel zum temporären Speichern von Datenwörtern, die zu der Empfangskomponente gesendet wurden. Die Sendekomponente läßt Datenwörter zu der Empfangskomponente strömen, ohne daß ein Bestätigungssignal notwendig ist, bis sie bestimmt, daß das Mittel zum temporären Speichern von Daten voll ist. Wenn die Empfangskomponente ein Datenwort von dem Mittel zum temporären Speichern von Daten liest, sendet sie ein Bestätigungssignal zu der Sendekomponente. Dieses Bestätigungssignal ist mit der Sendekomponente gekoppelt und kommuniziert dadurch mit einem Mittel zum Bestimmen dessen, ob das Mittel zum temporären Speichern von Daten voll ist, daß zusätzlicher Raum für ein Datenwort verfügbar ist.

**[0012]** Benötigt wird ein Flußsteuerungsprotokollsystem und -verfahren, das: (1) den Puffer in einem Empfangsrouter effektiv nutzt; (2) die Netzlatenz minimiert, die durch Steuersignale eingeschleppt wird; (3) die Erzeugung und das Senden des Steuersignals erzwingen kann; (4) ein Steuersignal mit anderen Rahmen multiplexiert, die zu demselben Port gesendet werden; (5) robust ist, d. h., das Sendefehler beheben kann; (6) an Senderahmen mit multiplen Prioritätsstufen anpaßbar ist; und (7) an Pufferbetriebsmittel anpaßbar ist, die in dem Empfänger vorgesehen sind.

#### ZUSAMMENFASSUNG DER ERFINDUNG

**[0013]** Die obigen Ziele werden durch ein Verfahren nach Anspruch 1 bzw. ein System nach Anspruch 18 erreicht. Bevorzugte Ausführungsformen sind in den Unteransprüchen angegeben.

**[0014]** Die Erfindung ist ein System und ein Verfahren zum Steuern des Sendens von Daten zwischen zwei Netzelementen. Ein erster Port eines Sendeelements ist mit einem zweiten Port eines Empfangselementes gekoppelt. Der zweite Port enthält Puffer zum temporären Speichern von Empfangsdaten, bis die Daten zu einem anderen Element gesendet werden können. Die vorliegende Erfindung enthält ein Register "Empfangen gegenwärtig voll" (RCFR), ein Register "Gesendet und nicht empfangen" (SANRR) und ein Register "Puffer belegt" (BBR) in dem Sendeelement. Das Sendeelement prüft sein BBR, um zu bestimmen, ob ein Puffer in dem Empfangselement verfügbar ist. Die Verfügbarkeit von Puffern kann unter Verwendung eines Protokolls mit einzelner Priorität oder eines Protokolls mit multipler Priorität bestimmt werden. Falls ein Puffer verfügbar ist, sendet das Sendeelement einen Datenrahmen zu dem Empfangselement und stellt das SANRR und das BBR ein, um anzugeben, daß ein Rahmen gesendet worden ist (und daß keine Bestätigung empfangen worden ist), daß der selektierte Puffer in dem Empfänger voll ist und daß keine zusätzlichen Datenrahmen zu diesem Puffer zu senden sind, bis der Puffer leer ist.

**[0015]** Wenn Daten durch das Empfangselement empfangen werden, werden sie zu einem verfügbaren Puffer gesendet. Wenn die Daten durch den Puffer empfangen werden, setzt das Empfangselement ein Bit in einem Register "Gegenwärtig voll" (CFR) und ein Bit in einem Register "Nächste zu sendende Meldung" (NMTSR). Jedes Bit ist dem Puffer zugeordnet. Jedes Bit in dem CFR gibt an, ob der zugeordnete Puffer leer ist. Jedes Bit in dem NMTSR gibt den Wert des CFR-Registers zu der Zeit an, zu der das vorherige Steuersignal gesendet wurde, mit der Ausnahme, daß dann, falls das CFR zu keiner Zeit zwischen Steuersignalen, d. h. Pufferstatusmeldungen, leer ist, das NMTSR gesetzt wird und auch dann nicht zurückgesetzt wird, wenn der Puffer leer ist, wenn das Steuersignal gesendet ist. Das Steuersignal kann nur das NMTSR oder sowohl das NMTSR als auch das CFR enthalten. Das Steuersignal wird auf derselben Signalleitung wie Daten gesendet, die gerade von dem Datenempfangselement zu dem Datensendeelement gesendet werden. Das Datenempfangselement kann die Rahmen, die in dieser Richtung gerade gesendet werden, mit dem Steuersignal multiplexieren.

**[0016]** Wenn das Sendeelement das Steuersignal empfängt, setzt es ein Bit zurück, das einem Puffer in dem SANRR zugeordnet ist, falls das Bit in dem SANRR gesetzt ist und falls das zugeordnete Bit in dem NMTSR-Abschnitt des Steuersignals gesetzt ist. Der Wert des RCFR wird entweder dem Wert des CFR oder des NMTSR in dem Steuersignal gleichgesetzt. Das Sendeelement setzt dann das zugeordnete Bit in dem BBR dem logischen ODER des SANRR und des RCFR gleich. Das Sendeelement wird Daten

nur dann senden, wenn ein Puffer in dem Empfangselement verfügbar ist, wie es durch das BBR angegeben ist, vorbehaltlich Datenprioritätsbeschränkungen, d. h., das Sendeelement wird erst einen Rahmen zu dem Puffer senden, wenn der Puffer leer ist und das RCFR gelöscht ist.

#### KURZE BESCHREIBUNG DER ZEICHNUNGEN

**[0017]** [Fig. 1](#) ist eine Darstellung eines Netzes, das Knoten, Router und Leitungen gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung enthält.

**[0018]** [Fig. 2](#) ist eine detailliertere Darstellung von zwei Routern in dem in [Fig. 1](#) gezeigten Netz gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung.

**[0019]** [Fig. 3](#) ist eine detailliertere Darstellung der zwei Router von [Fig. 2](#) gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung.

**[0020]** [Fig. 4](#) ist ein Zustandsdiagramm, das die Zustände von Puffern in einem Datensenderouter als Reaktion auf eine Pufferstatusmeldung, die Werte enthält, die zwei Register in dem Datenempfangsrouter darstellen, gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung zeigt.

**[0021]** [Fig. 5](#) ist ein Zustandsdiagramm, das die Zustände von Puffern in einem Datensenderouter als Reaktion auf eine Pufferstatusmeldung, die Werte enthält, die ein Register in dem Datenempfangsrouter darstellen, gemäß einer zweiten Ausführungsform der vorliegenden Erfindung zeigt.

**[0022]** [Fig. 6](#) ist ein Flußdiagramm der Sendeprotokolltechnik gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung.

**[0023]** [Fig. 7](#) ist ein Flußdiagramm einer Pufferstatusmeldungserzeugungs- und -sendetechnik gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung.

**[0024]** [Fig. 8](#) ist ein Flußdiagramm einer Pufferstatusmeldungempfangstechnik gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung.

**[0025]** [Fig. 9\(a\)–9\(g\)](#) sind Beispiele für Registerwerte gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung.

#### EINGEHENDE BESCHREIBUNG DER BEVORZUGTEN AUSFÜHRUNGSFORMEN

**[0026]** Unter Bezugnahme auf die Figuren, in denen gleiche Bezugszeichen identische oder funktionell ähnliche Elemente bezeichnen, wird nun eine bevor-

zugte Ausführungsform der vorliegenden Erfindung beschrieben. Ferner entspricht in den Figuren die Stelle am weitesten links von jedem Bezugszeichen der Figur, in der das Bezugszeichen zuerst verwendet wird.

**[0027]** Die vorliegende Erfindung ist ein Flußsteuerungsprotokollsystem und -verfahren, das: (1) die Puffer in einem Empfangsrouter effektiv nutzt; (2) die Netzlatenz minimiert, die durch Steuersignale eingeschleppt wird; (3) die Erzeugung und das Senden eines Steuersignals erzwingen kann; (4) ein Steuersignal mit anderen Datenrahmen multiplexiert, die zu demselben Port gesendet werden; (5) robust ist, d. h., eine Wiederherstellung nach Sendefehlern vornehmen kann; (6) an Senderahmen mit multiplen Prioritätsstufen anpaßbar ist; und (7) an Pufferbetriebsmittel anpaßbar ist, die in dem Empfänger vorgesehen sind.

**[0028]** [Fig. 1](#) ist eine Darstellung eines Netzes, das Knoten, Router und Leitungen gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung enthält. Das in [Fig. 1](#) dargestellte Netz enthält acht Knoten **102**, d. h. Knoten A–H **102A–H**, und sieben Router **104A–G**. In der bevorzugten Ausführungsform der vorliegenden Erfindung ist das Netz ein verteiltes Speichersystem. Das heißt, das System hat eine Architektur, die physisch verteilt ist, deren Speicher (Speicheranordnungen) aber logisch gemeinsam genutzt werden. Zum Beispiel kann ein Knoten, z. B. **102D**, den Zugriff auf eine Speicherstelle fordern, die der Knoten D **102D** für lokal hält, die physisch aber eigentlich an einer verschiedenen Stelle angeordnet ist und mit einem verschiedenen Knoten, z. B. dem Knoten A **102A**, gekoppelt ist. Der fordernde Knoten (Knoten D **102D**) identifiziert die Stelle des Speichers, und das Netz sucht die Daten schnell heraus. In einem verteilten Speichernetz ist die Latenz des Netzes ein wichtiger Leistungsparameter. Die Latenz des Netzes ist die Zeit, die benötigt wird, um einen fordernden Knoten mit den geforderten Daten zu beliefern, und zwar gemessen ab der Zeit, zu der die Speicherforderung gesendet wird.

**[0029]** Falls der Knoten D **102D** Daten fordert, die in dem Knoten A **102A** gespeichert sind, sendet der Knoten D **102D** eine Datenforderung durch das Netz. Die Datenforderung gelangt zum Beispiel von dem Knoten D **102D** zu einem dritten Router **104E**, zu einem zweiten Router **104B**, zu einem ersten Router **104A** und dann zu dem Knoten A **102A**. Die Daten werden aus der Speicheranordnung, die mit dem Knoten A **102A** gekoppelt ist, herausgesucht und durch das Netz zu dem Knoten D **102D** gesendet. Die Daten werden, wie oben beschrieben, in Rahmen durch das Netz gesendet. Der Knoten, z. B. der Knoten A **102A**, bestimmt den Routenweg durch das Netz und speichert diese Routeninformationen in dem Rahmenkopf jedes Rahmens. Zum Beispiel

können die Daten, die gerade von dem Knoten A **102A** zu dem Knoten D **102D** gesendet werden, Routeninformationen haben, die jeden Rahmen anzeigen, von dem Knoten A **102A** zu dem ersten Router **104A**, dann von dem ersten Router **104A** zu dem zweiten Router **104B**, dann von dem zweiten Router **104B** zu dem dritten Router **104E** und dann von dem dritten Router **104E** zu dem Knoten D **102D** zu gelangen. Da die Latenz des Netzes die Dauer zwischen dem Senden einer Datenforderung durch den fordernden Knoten und dem Empfang der Daten durch den fordernden Knoten ist, wird durch das Reduzieren der Verzögerung, die durch jeden der Router **104** eingeschleppt wird, die Netzlatenz reduziert. Jeder Router **104** enthält Puffer, wie oben beschrieben, zum temporären Speichern der Daten, die empfangen werden. In der bevorzugten Ausführungsform sind sechs Puffer pro Router-Port und sechs Ports auf einem Router vorhanden. Die Größe jedes Puffers ist der maximalen Rahmenlänge wie z. B. 1292 Bits gleich. Wenn ein Router **104** bezeichnet wird, spielen jedoch die Kosten eine wichtige Rolle. Das Erhöhen der Anzahl der Puffer in jedem Router ist teuer und in manchen Situationen auf Grund von Größenbeschränkungen auf einer integrierten Schaltung, auf der die Puffer liegen, nicht machbar.

**[0030]** Die Router sind durch Duplexsignalleitungen **106** miteinander verbunden. Jede Duplexsignalleitung **106A** ist dazu in der Lage, Daten in beiden Richtungen gleichzeitig zu transportieren, z. B. sowohl von dem zweiten Port des ersten Routers **104A** als auch zu dem zweiten Port des ersten Routers **104A**. Die Duplexsignalleitung **106A** ist unten unter Bezugnahme auf [Fig. 2](#) eingehender beschrieben.

**[0031]** [Fig. 2](#) ist eine detailliertere Darstellung von zwei Routern **104** in dem Netz, das in [Fig. 1](#) gezeigt ist, gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung. Die Duplexsignalleitung **106A** ist durch zwei Signalleitungen **210, 212** dargestellt. Jede Signalleitung ist unidirektional. Eine erste Signalleitung **210** transportiert Informationen von dem Port 1 **202A** des ersten Routers **104A** zu dem Port 6 **202B** des zweiten Routers **104B**. Eine zweite Signalleitung **212** transportiert Informationen von dem Port 6 **202B** des zweiten Routers **104B** zu dem Port 2 **202A** des ersten Routers **104A**. In der bevorzugten Ausführungsform enthält jede dargestellte Signalleitung **210, 212** 43 Signalleitungen, die Daten zwischen Ports parallel transportieren. Jeder Router **104** enthält sechs Ports **202**, die jeweils mit einem Kreuzschienenschalter **206** gekoppelt sind. Zusätzlich ist jeder Port mit sechs Arbitern **204**, z. B. A1–A6, verbunden, und jeder Arbeiter **204** ist einem Ausgangsabschnitt von einem der Ports zugeordnet. Zum Beispiel ist der Arbeiter 2 **204A** auf dem ersten Router **104A** mit einem Ausgangsabschnitt von Port 2 **202A** gekoppelt. Der Ausgangsabschnitt von Port 2 **202A** ist mit der Signalleitung **210** gekoppelt. Jeder

Port, z. B. der Port 2 **202A** des ersten Routers **104A** und der Port 6 **202B** des zweiten Routers **104B**, ist, wie in [Fig. 2](#) gezeigt, mit nur einem anderen Port in einem verschiedenen Router gekoppelt. Zum Beispiel ist der Port 2 **202A** des ersten Routers **104A** mit dem Port 6 **202B** des zweiten Routers **104B** gekoppelt, d. h., der Port 2 **202A** des ersten Routers **104A** empfängt Daten nur von und sendet Daten nur zu dem Port 6 **202B** des zweiten Routers **104B**, und der Port 6 **202B** des zweiten Routers **104B** empfängt Daten nur von und sendet Daten nur zu dem Port 2 **202A** des ersten Routers **104A**. Die Operation der Arbiter **204**, der Kreuzschienenschalter **206** und der Ports **202** werden unten unter Bezugnahme auf die [Fig. 3–Fig. 9](#) eingehender beschrieben.

**[0032]** [Fig. 3](#) ist eine detailliertere Darstellung des ersten Routers **104A** und des zweiten Routers **104B**, die in den [Fig. 1](#) und [Fig. 2](#) gezeigt sind, gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung. Jeder Port **202** in jedem Router **104** enthält einen Flußsteuerungsempfänger **304**, ein Pufferarray **322**, eine Modifizierungseinheit **326**, einen Multiplexer (MUX) **328**, einen Datensynchronisierer **324** und einen Flußsteuerungssender **305**. Jeder Flußsteuerungsempfänger **304** enthält einen Pufferstatusmeldung-(BSM)-Empfänger **306**, der mit einem Register "Gesendet und nicht empfangen" (SANRR) **308**, einem Register "Empfangen gegenwärtig voll" (RCFR) **310** und einem Register "Puffer belegt" (BBR) **312** verbunden ist. Jeder Flußsteuerungssender **305** enthält einen BSM-Generator **316**, der mit einem Register "Nächste zu sendende Meldung" (NMTSR) **318** und einem Register "Gegenwärtig voll" (CFR) **320** verbunden ist. Obwohl es unter Bezugnahme auf die Router **104** beschrieben ist, enthält jeder Knoten **102** auch die Register in dem Flußsteuerungsempfänger **304A** und dem Flußsteuerungssender **305B**. Die Operation dieser Portkomponenten wird nun unter Bezugnahme auf das obenerwähnte Datenübertragungsbeispiel beschrieben.

**[0033]** Falls der Knoten D **102D** Daten fordert, die in einer Speicheranordnung gespeichert sind, die durch den Knoten A **102A** gesteuert wird, sendet der Knoten D **102D** eine Forderung an Knoten A **102A**. Der Knoten A **102A** sucht die Daten heraus, bestimmt die Route, die die Datenrahmen durch das Netz nehmen werden, und speichert diese Routeninformationen in dem Kopf jedes Rahmens, wie oben beschrieben. Der erste Router **104A** empfängt Daten in Form von Rahmen von dem Knoten A **102A**. Der Knoten A **102A** ist mit Port 3 des ersten Routers **104A** gekoppelt. Daher wird jeder Rahmen zu einem Puffer in dem Pufferarray **322A** von Port 3 gesendet. Die Routeninformationen in dem Rahmenkopf identifizieren den nächsten Router **104**, den die Daten durchlaufen müssen. Bei dem vorliegenden Beispiel ist der nächste Router der zweite Router **104B**. Port 2 **202A** des ersten Routers **104A** ist mit Port 6 **202B** des

zweiten Routers **104B** gekoppelt. Daher identifizieren die Routeninformationen in dem Rahmenkopf den Port 2 **202A** als Ausgangsport für den ersten Router **104A**. Diese Ausgangsportinformationen werden durch den Arbiter **204** empfangen, der dem Ausgangsport zugeordnet ist. Bei dem vorliegenden Beispiel ist der Arbiter 2 **204A** dem Port 2 **202A** des ersten Routers **104A** zugeordnet. Der Arbiter 2 **204A** empfängt Signale von dem Flußsteuerungsempfänger **304A**, die angeben, ob eine Datenübertragung zu dem Empfangsport erfolgen kann. Bei dem vorliegenden Beispiel ist der Empfangsport der Port 6 **202B** des zweiten Routers **104B**. Der Arbiter 2 **204A** sendet dann ein Signal zu einem ersten Puffer in dem Pufferarray **322A** von Port 3, das das Pufferarray **322A** von Port 3 instruiert, die Daten in dem ersten Puffer zu der Kreuzschienennmatrix **206** zu senden. Der Arbiter 2 **204A** sendet auch ein Signal zu der Kreuzschienennmatrix **206**, das bewirkt, daß die Kreuzschienennmatrix **206** den ersten Puffer in dem Pufferarray **322A** von Port 3 mit Port 2 **202A** des ersten Routers **104A** koppelt. Die Operation von jedem Arbiter **204** ist eingehender beschrieben in der gleichzeitig schwebenden Patentanmeldung mit dem Titel "METHOD AND APPARATUS FOR COORDINATING ACCESS TO AN OUTPUT OF A ROUTING DEVICE IN A PACKET SWITCHING NETWORK", eingereicht am 22. Februar 1996 durch Jeffrey D. Larson, Albert Mu und Thomas M. Wicki, die in ihrer Gesamtheit oben durch Bezugnahme identifiziert und inkorporiert ist. Ähnlich ist die Operation von jeder Kreuzschienennmatrix **206** beschrieben in der ebenfalls schwebenden Patentanmeldung mit dem Titel "CROSSBAR SWITCH AND METHOD WITH REDUCED VOLTAGE SWING AND NO INTERNAL BLOCKING DATA PATH", eingereicht am 22. Februar 1996 durch Albert Mu und Jeffrey D. Larson, die in ihrer Gesamtheit oben durch Bezugnahme identifiziert und inkorporiert ist.

**[0034]** Bevor das Signal zu dem Arbiter 2 **204A** gesendet wird, das angibt, ob eine Datenübertragung zu dem Empfangsport (Port 6 **202B** des zweiten Routers **104B**) erfolgen kann, sendet der Flußsteuerungsempfänger **304A** ein Signal zu der Modifizierungseinheit **326A**, das die Adresse eines leeren Puffers (des Empfangspuffers) in dem Pufferarray **322B** von Port 6 identifiziert. Das BBR **312A** in dem Flußsteuerungsempfänger **304A** enthält ein Bit für jeden Puffer in dem Pufferarray **322B** von Port 6 des zweiten Routers **104B**. Der Wert von jedem Bit in dem BBR **312A** gibt an, ob der erste Router **104A** bestimmt hat, daß der zugeordnete Puffer in dem Pufferarray **322B** von Port 6 des zweiten Routers **104B** leer ist. Die Technik zum Setzen und Zurücksetzen der Bits in dem BBR **312A** ist unten unter Bezugnahme auf die [Fig. 4–Fig. 9](#) beschrieben. Die Modifizierungseinheit **326A** in dem ersten Router **104A** empfängt die Adresse eines leeren Puffers in dem Pufferarray **322B** von Port 6 in dem zweiten Router **104B**.

Dieser leere Puffer ist der Empfangspuffer.

**[0035]** Wenn der Arbiter 2 **204A** den Kreuzschienenschalter **206A** instruiert, den ersten Puffer in dem Pufferarray **322A** von Port 3 mit der Modifizierungseinheit **326** zu verbinden, sendet der Arbiter 2 **204A** auch ein Signal zu dem Flußsteuerungsempfänger **304A**, das den Flußsteuerungsempfänger **304A** darüber informiert, daß ein Rahmen gesendet worden ist. Der BSM-Empfänger **306A** in dem Flußsteuerungsempfänger **304A** modifiziert den Wert des Bits in dem SANRR **308A**, das dem Empfangspuffer in dem Pufferarray **322B** von Port 6 zugeordnet ist, um anzugeben, daß ein Rahmen zu dem Empfangspuffer gesendet worden ist und daß der BSM-Empfänger **306A** ein Signal, das den Empfang des Rahmens bestätigt, nicht empfangen hat. Der Wert des BBR **312** ist, wie unten beschrieben, dem logischen ODER des SANRR **308A** und des RCFR **310A** gleich. Da der Wert des Bits, das dem Empfangspuffer in dem Pufferarray **322B** von Port 6 zugeordnet ist, gleich einer logischen Eins gesetzt wird, wird das zugeordnete Bit in dem BBR **312A** auch gleich einer logischen Eins gesetzt. Wenn dieses Bit einer logischen Eins gleich ist, wird das Pufferadressensignal, das von dem Flußsteuerungsempfänger **304A** zu der Modifizierungseinheit **326A** in dem ersten Router **104A** gesendet wird, den Empfangspuffer erst identifizieren, wenn dieses Bit in dem BBR **312A** zurückgesetzt ist, d. h., der Empfangspuffer wird als "belegt" identifiziert, bis das BBR **312A** zurückgesetzt ist. Falls alle Puffer in dem Pufferarray **322B** von Port 6 nicht leer sind, wie es durch die Informationen in der neuesten BSM bestimmt wird oder wie es durch den Flußsteuerungsempfänger **304A** durch Setzen des SANRR **308A** und BBR **312A** bestimmt wird, wenn der erste Router **104A** einen Rahmen zu dem zweiten Router sendet, werden dann alle Bits in dem BBR **312A** gleich einer logischen Eins gesetzt werden. In dieser Situation benachrichtigt der BSM-Empfänger **306A** den Arbiter 2 **204A**, und der Arbiter 2 **204A** wird das Senden von jeglichen zusätzlichen Rahmen zu dem Port 6 **202B** nicht gestatten, bis wenigstens ein Bit in dem BBR **312A** zurückgesetzt ist. Die obige Beschreibung erläutert die Umgebung, wenn nur eine einzelne Transportprioritätsstufe für alle Rahmen existiert. Die vorliegende Erfindung unterstützt auch Rahmen, wie unten beschrieben, die multiple Transportprioritätsstufen haben.

**[0036]** Die Modifizierungseinheit **326A** empfängt den Rahmen von dem Pufferarray **322A** von Port 3 über den Kreuzschienenschalter **206A** und empfängt die Empfangspufferadresse von dem Flußsteuerungsempfänger **304A**. Die Modifizierungseinheit **326A** modifiziert den Rahmenkopf zum Beispiel durch Entfernen der Routeninformationen, die Port 2 **202A** des zweiten Routers **104B** identifizieren, und verschiebt die verbleibenden Routeninformationen in dem Rahmenkopf. Die Modifizierungseinheit **326A**

fügt auch die Empfangspufferadresse zu dem Rahmenkopf hinzu. Die Modifizierungseinheit **326A** sendet den Rahmen zu dem MUX **328A**, der den Rahmen über die Signalleitung **210** der Duplexsignalleitungen **106A** zu dem zweiten Router **104B** sendet.

**[0037]** Ein Datensynchronisierer **324B** in dem zweiten Router **104B** empfängt die Daten von Port 2 **202A** des ersten Routers **104A**. Jeder Router **104** enthält einen Takt (nicht gezeigt). Eine Funktion des Datensynchronisierers **324B** ist das Synchronisieren der empfangenen Daten mit dem internen Takt des zweiten Routers **104B**. Eine andere Funktion des Datensynchronisierers **324B** ist das Unterscheiden zwischen Datenrahmen und Pufferstatusmeldungen, die von dem ersten Router **104A** empfangen werden, unter Verwendung eines Daten/Status-Steuersignals. Nachdem bestimmt worden ist, daß der empfangene Rahmen ein Datenrahmen ist, sendet der Datensynchronisierer **324B** die Daten zu dem Empfangspuffer in dem Pufferarray **322B** von Port 6. Nach dem Empfang des Rahmenkopfes kann der zweite Router mit dem Senden der Daten zu dem dritten Router **104E** unter Verwendung der oben beschriebenen Technik beginnen. Falls jedoch alle Puffer in dem Pufferarray von Port 4 in dem dritten Router **104E** nicht leer sind, wird der zweite Router **104B** das Speichern der Daten, die von dem ersten Router **104A** empfangen werden, in dem Pufferarray **322B** von Port 6 fortsetzen.

**[0038]** Wenn Daten durch den Empfangspuffer in dem Pufferarray **322B** von Port 6 empfangen werden, wird der BSM-Generator **316B** in dem BSM-Sender **305B** benachrichtigt. Der BSM-Generator **316B** setzt ein Bit in dem NMTSR **318B**, das dem Empfangspuffer zugeordnet ist, gleich einer logischen Eins, um anzugeben, daß ein Abschnitt des Rahmens durch den Empfangspuffer empfangen worden ist. Zusätzlich setzt der BSM-Generator **316B** ein zugeordnetes Bit in dem CFR **320B** gleich einer logischen Eins, um anzugeben, daß der Empfangspuffer nicht leer ist. Der BSM-Sender **305B** erzeugt und sendet eine BSM. Der Inhalt der BSM ist unten unter Bezugnahme auf die [Fig. 4 - Fig. 9](#) beschrieben. Die BSM enthält eine Bestätigung, daß der durch den ersten Router **104A** gesendete Rahmen empfangen wurde, und enthält auch eine Angabe des Status der Puffer in dem Pufferarray von Port 6, z. B. ob jeder einzelne Puffer leer ist. Der BSM-Generator **316B** sendet die Daten zu einem Eingangsport des MUX **328B** in dem zweiten Router **104B**. Zusätzlich kann der BSM-Generator **316B** ein Signal an den Arbiter 6 **204B** senden, das das Senden einer BSM an den ersten Router **104A** fordert. Der Arbiter 6 **204** erzeugt ein Signal, das den MUX **328B** steuert. Wenn dieses Signal durch den Arbiter 6 **204B** gesendet ist, wird die BSM über die Signalleitung **212** zu dem Port 2 **202A** des ersten Routers **104A** gesendet.

**[0039]** Der Datensynchronisierer **324A** in Port 2 **202A** empfängt die BSM. Der Datensynchronisierer **324A** bestimmt, daß die Empfangsdaten eine BSM sind, und sendet die Daten an den BSM-Empfänger **306A** in dem Flußsteuerungsempfänger **304A**. Der Flußsteuerungsempfänger **304A** aktualisiert das SANRR **308A**, RCFR **310A** und BBR **312A** auf der Basis der empfangenen BSM. Die Bitwerte in diesen Registern werden durch den BSM-Empfänger **306A** empfangen, wie oben beschrieben. Der BSM-Empfänger **306A** bestimmt, ob noch mehr Rahmen zu dem Port 6 **202B** in dem zweiten Router **104B** gesendet werden können. Die Operation des Flußsteuerungsempfängers **304A** und des Flußsteuerungssenders **305B** ist unten unter Bezugnahme auf die [Fig. 4–Fig. 9](#) eingehender beschrieben.

**[0040]** [Fig. 4](#) ist ein Zustandsdiagramm, das die Zustände von Puffern in dem Flußsteuerungsempfänger **304A** in dem ersten Router **104A** als Reaktion auf eine BSM, die Werte enthält, die sowohl das NMTSR **318B** als auch das CFR **320B** in dem Flußsteuerungssender **305B** in dem zweiten Router **104B** darstellen, gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung zeigt. Ein Vorteil des Flußsteuerungsprotokolls der vorliegenden Erfindung ist der, daß es robust ist. Das heißt, das Flußsteuerungsprotokoll kann eine Wiederherstellung nach Sendefehlern wie z. B. von verlorengegangenen BSMs oder verlorengegangenen Datenrahmen vornehmen. Wenn Rahmen von dem ersten Router **104A** zu dem zweiten Router **104B** gesendet werden, nutzt das Flußsteuerungsprotokoll der vorliegenden Erfindung fünf Register. Diese Register sind das SANRR **308A**, das RCFR **310A** und das BBR **312A** in dem Flußsteuerungsempfänger **304A** in dem ersten Router **104A** und das NMTSR **318B** und das CFR **320B** in dem Flußsteuerungssender **305B** in dem zweiten Router **104B**. Die folgende Beschreibung des Flußsteuerungsprotokolls nimmt auf diese fünf Register Bezug. Jedes dieser fünf Register enthält, wie oben beschrieben, ein Bit, das jedem Puffer in dem Pufferarray **322B** von Port 6 zugeordnet ist. In der bevorzugten Ausführungsform enthält das Pufferarray **322B** von Port 6 sechs Puffer. [Fig. 4](#) und [Fig. 5](#) zeigen das Zustandsdiagramm für ein Bit in jedem dieser fünf Register. Daher wird jetzt das Flußsteuerungsprotokoll für einen einzelnen Puffer erläutert.

**[0041]** Wenn das Netz initialisiert wird, werden die Register in dem Flußsteuerungsempfänger **304A** auf die Werte initialisiert, die Zustand III zugeordnet sind. Das heißt, RCFR = 1, SANRR = 0 und BBR = 1. Der BSM-Empfänger **306A** setzt den Wert des BBR **312A**, wie oben beschrieben, gleich dem logischen ODER des RCFR **310A** und des SANRR **308A**. Die Flußsteuerungsempfängerregister, d. h. SANRR **308A**, RCFR **310A** und BBR **312A**, werden auf Werte gesetzt, die bei Zustand III gezeigt sind, da dies die konservativste Flußsteuerungsannahme ist. Das

heißt, wenn die Flußsteuerungsempfängerregister im Zustand III sind, wird der Flußsteuerungsempfänger **304A** verhindern, daß irgendwelche Daten zu Port 6 **202B** des zweiten Routers **104A** gesendet werden, da das BBR **312A**, wenn es im Zustand III ist, gleich einer logischen Eins gesetzt ist, die angibt, daß der Empfangspuffer nicht leer ist. Falls der Empfangspuffer leer ist, modifiziert das Flußsteuerungsprotokoll, wie unten beschrieben, den Zustand der Flußsteuerungsempfängerregister, um den richtigen Zustand anzugeben. Die BSM enthält, wie oben beschrieben, die Bitwerte des NMTSR **318B** und des CFR **320B**. Bei diesem Beispiel gibt eine BSM = 1,0 an, daß das Bit in dem NMTSR **318B**, das dem Empfangspuffer zugeordnet ist, einen Wert gleich einer logischen Eins hat und daß ein Bit in dem CFR **320B**, das dem Empfangspuffer zugeordnet ist, einen Wert gleich einer logischen Null hat. Nach dem Übergang von dem Zustand II in den Zustand III gibt es nur zwei zulässige BSM-Werte, d. h. BSM = 1,1 und BSM = 1,0. Der Grund dafür ist der, daß das NMTSR **318B** entweder: (1) dem Wert des CFR **320B** gleich ist, der in der vorherigen BSM gesendet wurde; oder (2) einer logischen Eins für wenigstens eine BSM gleich ist, wenn ein Rahmen durch den Empfangspuffer empfangen wird, ungeachtet des Wertes des zugeordneten Bits in dem CFR **210B** in der vorherigen BSM. Wenn es im Zustand III ist, ist der Wert des CFR **320B** in der letzten BSM gleich einer logischen Eins, da das RCFR **310A** diesen Wert speichert. Daher muß der Wert für das NMTSR **318B** in der nächsten BSM auf der Basis von (1) und (2) oben gleich einer logischen Eins sein. Falls die Register jedoch als Resultat einer Initialisierungsprozedur im Zustand III sind, ist eine BSM = 0,0 gültig, was dazu führt, daß in den Registern in dem Flußsteuerungsempfänger **304A** ein Übergang von dem Zustand III in den Zustand I erfolgt.

**[0042]** Eine BSM = 1,1 gibt an, daß der Empfangspuffer nicht leer ist. Falls eine BSM = 1,1 durch den Flußsteuerungsempfänger **304A** empfangen wird, bleibt daher der Zustand der Register in dem Flußsteuerungsempfänger **304A** derselbe. Eine BSM = 1,0 gibt an, daß der Rahmen durch den Empfangspuffer in dem Pufferarray **322B** von Port 6 empfangen worden ist und zu dem dritten Router **104E** gesendet worden ist. Deshalb ist der Empfangspuffer nicht leer. Falls eine BSM = 1,0 empfangen wird, gehen die Register in dem Flußsteuerungsempfänger **304A** von dem Zustand III in den Zustand I über.

**[0043]** In dem Zustand I sind das SANRR **308A**, RCFR **310A** und BBR **312A** alle gleich einer logischen Null. Da das BBR **312A** gleich einer logischen Null ist, teilt der Flußsteuerungsempfänger **304A** dem Arbiter 2 **204A** mit, daß der Empfangspuffer leer ist. Vorbehaltlich Prioritätsbeschränkungen, die unten erläutert sind, instruiert dann, falls ein Rahmen in Port 3 **202A** verfügbar ist, um zu Port 6 **202B** des

zweiten Routers **104B** gesendet zu werden, der Arbitter 2 **204A** die Kreuzschienenmatrix **206A**, um den Sendepuffer in dem Pufferarray **322A** von Port 3 mit der Modifizierungseinheit **326A** von Port 2 **202A** zu verbinden. Der Arbitter 2 **204A** sendet auch, wie oben beschrieben, ein Signal zu dem Flußsteuerungsempfänger **304A**, das dem Flußsteuerungsempfänger **304A** mitteilt, daß die Daten gesendet worden sind. Zu dieser Zeit setzt der BSM-Empfänger **306A** das SANRR **308A** gleich einer logischen Eins, und deshalb setzt er das BBR **312B** gleich einer logischen Eins. Das RCFR **310A** wird nicht modifiziert. Falls Daten zu dem zweiten Router **104B** gesendet werden, gehen deshalb die Register in dem Flußsteuerungsempfänger **304A** in den Zustand II über. Falls keine Daten gesendet werden, während sie in dem Zustand I sind, bleiben die Register in dem Flußsteuerungsempfänger **304A** im Zustand I.

**[0044]** Im Zustand II sind das SANRR **308A** und das BBR **312A** gleich einer logischen Eins, und das RCFR **310A** ist gleich einer logischen Null. In diesem Zustand geben die Register in dem Flußsteuerungsempfänger **304A** über eine BSM an, daß ein Rahmen zu dem Empfangspuffer in dem Pufferarray **322B** von Port 6 gesendet worden ist, aber der Flußsteuerungsempfänger **304A** keine Bestätigung empfangen hat, daß der Empfangspuffer den Rahmen empfangen hat. Zusätzlich ist das RCFR **310A** gleich einer logischen Null, die angibt, daß der Empfangspuffer auf der Basis der neuesten BSM leer ist. Wenn sie in dem Zustand II sind, wird der Flußsteuerungsempfänger **304A** nicht zulassen, daß ein anderer Rahmen zu dem Empfangspuffer gesendet wird. Falls eine BSM = 0,0 empfangen wird, bleiben die Register in dem Flußsteuerungsempfänger **304A** im Zustand II, da die BSM angibt, daß keine Daten in dem Empfangspuffer empfangen wurden, da die vorherige BSM gesendet wurde. Falls eine BSM = 1,1 empfangen wird, gehen die Register in dem Flußsteuerungsempfänger **304A** in den Zustand III über. Eine BSM = 1,1 gibt an, daß einige Rahmendaten durch das Empfangsregister empfangen worden sind und daß das Empfangsregister nicht leer ist. Da einige Rahmendaten empfangen worden sind, wird das SANRR **308A** auf eine logische Null zurückgesetzt. Da der Empfangspuffer nicht leer ist, wird das RCFR **310A** auf eine logische Eins gesetzt. Eine Erläuterung des Zustandes III ist oben vorgesehen. Falls eine BSM = 1,0 empfangen wird, gehen die Register in dem Flußsteuerungsempfänger **304A** in den Zustand I über. Wenn sie in dem Zustand II sind, gibt eine BSM = 1,0 an, daß Rahmendaten durch den Empfangspuffer empfangen worden sind und daß der Rahmen zu dem dritten Router **104E** gesendet worden ist. Deshalb ist der Empfangspuffer leer. Da der Rahmen durch den Empfangspuffer empfangen worden ist, wird das SANRR **308A** zurückgesetzt. Da der Empfangspuffer leer ist, bleibt das RCFR **310A** einer logischen Eins gleich. Wenn sie in dem Zustand I sind, kann der

Flußsteuerungsempfänger **304A** den Arbitter 2 **204A** instruieren, einen neuen Rahmen zu dem Empfangspuffer in dem Pufferarray **322B** von Port 6 zu senden.

**[0045]** Das Flußsteuerungsprotokollsystem und -verfahren der vorliegenden Erfindung ist robust, wie oben erwähnt. Das heißt, das Protokoll kann eine Wiederherstellung nach Sendefehlern vornehmen. Ein Zeitüberschreitungsmerkmal des Flußsteuerungsprotokolls ermöglicht die Wiederherstellung nach BSM- oder Rahmensexdefehlern durch das Netz, wodurch sonst die Anzahl von verfügbaren Puffern in den Pufferarrays **322** verringert werden könnte. Falls die Register in dem Flußsteuerungsempfänger **304A** im Zustand II sind und der Flußsteuerungssender **305B** in dem zweiten Router **104B** zum Beispiel eine BSM = 1,0 erzeugt, gehen die Register in dem Flußsteuerungsempfänger **304A** in den Zustand I über und steht der Empfangspuffer zur Verfügung, um zusätzliche Rahmen zu empfangen, wie oben angegeben. Falls diese BSM durch den Flußsteuerungsempfänger **304A** jedoch nie empfangen wird, tritt ein Problem auf. Die Register in dem Flußsteuerungsempfänger **304A** sind im Zustand II, aber der Flußsteuerungssender **305B** nimmt an, daß die Register in dem Flußsteuerungsempfänger **304A** im Zustand I sind. Das heißt, der Flußsteuerungssender **305B** erzeugte und sendete eine BSM (BSM = 1,0), die angibt, daß der Empfangspuffer den gesendeten Rahmen empfing und daß der Empfangspuffer zur Verfügung steht, um zusätzliche Rahmen zu empfangen. Da der Empfangspuffer leer ist und keine neuen Rahmen gesendet worden sind, erzeugt der Flußsteuerungssender **305B** nur BSMs, die der BSM = 0,0 gleich sind. Da die Register in dem Flußsteuerungsempfänger **304A** im Zustand II sind, wird eine BSM = 0,0 den Zustand dieser Register nicht verändern. Daher wird der Flußsteuerungsempfänger **304A** in dem ersten Router **104A** den Empfangspuffer nicht mehr verwenden. Falls ähnlich der Rahmen, der von dem ersten Router **104A** gesendet wird, durch den zweiten Router **104B** nie empfangen wird, sind die Register in dem Flußsteuerungsempfänger **304A** im Zustand II, während der Flußsteuerungssender **305B** in dem zweiten Router **104B** nur BSMs erzeugt, die der BSM = 0,0 gleich sind, die angibt, daß keine Rahmen empfangen worden sind.

**[0046]** Puffer sind, wie oben beschrieben, teure und kostbare Betriebsmittel. Durch den Verlust eines einzigen Puffers wird die Effektivität des Netzes verringert. Das Flußsteuerungsprotokoll der vorliegenden Erfindung führt einen Zeitüberschreitungsmechanismus ein, um den Verlust von Puffern zu verhindern. Der Zeitüberschreitungsmechanismus bestimmt, ob eine vorbestimmte Mindestmenge an Taktzyklen aufgetreten ist, während die Register in dem Flußsteuerungsempfänger **304A** im Zustand II sind. Falls die Register für wenigstens diese Mindestlänge der Zeit im Zustand II sind, versetzt der Fluß-

steuerungsempfänger **304A** dann seine Register zurück in den Zustand I und inkrementiert einen Fehlerzähler (nicht gezeigt), um zu identifizieren, daß ein Fehler aufgetreten ist. Eine Technik zum Bestimmen, ob die vorbestimmte Anzahl von Taktzyklen aufgetreten ist, ist das Nutzen eines Zählers für jeden Puffer in jedem Pufferarray **322**. Falls zum Beispiel der Zähler für den Empfangspuffer abläuft, versetzt der Flußsteuerungsempfänger **304A** seine Register vom Zustand II in den Zustand I. Die Anzahl von Bits, die für jeden Zähler erforderlich ist, kann jedoch ungefähr zehn oder elf Bits betragen. Wenn jeder Router **104** einen Elf-Bit-Zähler für jeden zugeordneten Puffer haben muß, werden die Kosten erhöht und wird die potentielle Leistung der Router **104** verringert. In der bevorzugten Ausführungsform können bis zu 36 Puffer jedem Router **104** zugeordnet sein. Die vorliegende Erfindung löst das Problem, indem ein einziger Zehn-Bit-Zähler für alle Puffer, die einem besonderen Router zugeordnet sind, verwendet wird und dann nur ein Ein-Bit-Flag für jeden der zugeordneten Puffer verwendet wird. Der Zehn-Bit-Zähler zählt die Taktzyklen kontinuierlich, und wenn der Zehn-Bit-Zähler zum Beispiel von den ganzen logischen Einsen zu den ganzen logischen Nullen übergeht, wird das Ein-Bit-Flag auf eine logische Eins für jeden Puffer gesetzt, dessen zugeordnete Register in dem Flußsteuerungsempfänger **104A** im Zustand II sind. Falls die Register in dem Flußsteuerungsempfänger **104A** den Zustand wechseln, wird das Ein-Bit-Flag auf eine logische Null zurückgesetzt. Falls der Zustand der Register jedoch bis zum nächsten Übergang des Zehn-Bit-Zählers zu den ganzen logischen Nullen nicht wechselt, tritt eine Zeitüberschreitung auf. Wenn eine Zeitüberschreitung auftritt, versetzt der Flußsteuerungsempfänger **104A** seine Register, wie oben beschrieben, in den Zustand I.

**[0047]** [Fig. 5](#) ist ein Zustandsdiagramm, das die Zustände von Puffern in dem Flußsteuerungsempfänger **304A** in dem ersten Router **104A** als Reaktion auf eine BSM, die Werte enthält, die nur das NMTSR **318B** in dem Flußsteuerungssender **305B** des zweiten Routers **104B** darstellen, gemäß einer zweiten Ausführungsform der vorliegenden Erfindung zeigt. Eine Erläuterung dazu, wie der Flußsteuerungsempfänger **304A** den Zustand I, den Zustand II und den Zustand III interpretiert, ist oben unter Bezugnahme auf [Fig. 4](#) enthalten. Falls die BSM nur den Wert in dem NMTSR **318B** darstellt, interpretiert das Flußsteuerungsprotokoll der vorliegenden Erfindung die BSM anders als die Situation, wenn die BSM die Werte sowohl des NMTSR **318B** als auch des CFR **320B** darstellt. Wenn der Flußsteuerungsempfänger **304A** initialisiert wird, werden die Register in dem Flußsteuerungsempfänger **304A** auf die Werte gesetzt, die beim Zustand III angegeben sind. Falls das Bit, das dem Empfangsregister in dem NMTSR **318B** zugeordnet ist, gleich einer logischen Eins ist, als die BSM erzeugt wurde, hat die BSM dann einen Wert gleich

einer logischen Eins, d. h. BSM = 1. Eine BSM = 1 gibt an, daß entweder (1) Rahmendaten durch den Empfangspuffer empfangen worden sind, da die vorherige BSM gesendet wurde, oder (2) der Empfangspuffer zu der Zeit des Sendens der vorherigen BSM nicht leer war, wie es durch das CFR **320B** dargestellt wird. Falls eine BSM = 1 durch den Flußsteuerungsempfänger **304A** empfangen wird, wenn seine Register im Zustand III sind, ändert sich der Zustand dieser Register nicht. Eine BSM = 0 gibt an, daß der Empfangspuffer leer ist. Wenn eine BSM = 0 durch den Flußsteuerungsempfänger **304A** empfangen wird, kann der Zustand seiner Register von dem Zustand III in den Zustand I übergehen, der angibt, daß ein anderer Rahmen zu dem Empfangspuffer gesendet werden kann.

**[0048]** Der Zustand der Register in den Flußsteuerungsempfängern **104A** bleibt der Zustand I, bis ein Rahmen zu dem Empfangspuffer in dem Pufferarray **322B** von Port 6 gesendet wird. Wenn ein Rahmen gesendet wird, geht der Zustand der Register in dem Flußsteuerungsempfänger **104A** von dem Zustand I in den Zustand II über. Falls eine BSM = 0 während des Zustandes II empfangen wird, tritt keine Zustandsänderung auf. Eine BSM = 0 gibt an, daß seit dem Senden der vorherigen BSM keine Daten durch den Empfangspuffer empfangen wurden. Eine BSM = 1 ist eine Bestätigung, daß einige Rahmendaten durch den Empfangspuffer seit dem Senden der vorherigen BSM empfangen wurden. Falls eine BSM = 1 empfangen wird, wird das SANRR **308A** deshalb zurückgesetzt, da der Rahmen durch den Empfangspuffer empfangen wurde. Da das SANRR **308A** jedoch für wenigstens eine BSM gleich 1 gesetzt sein muß, nachdem der Empfangspuffer Daten empfängt, können keine Informationen darüber, ob der Empfangspuffer zu der Zeit, zu der die BSM gesendet wurde, leer ist, wahrgenommen werden, wenn eine BSM = 1 empfangen wird, wenn die Register in dem Flußsteuerungsempfänger **304A** im Zustand II sind. Daher wird das zugeordnete Bit in dem SANRR **308A** zurückgesetzt und wird das zugeordnete Bit in dem RCFR **310A** auf eine logische Eins gesetzt. Wenn die BSM nur das NMTSR **318B** darstellt, gehen die Register in dem Flußsteuerungsempfänger **304A** als Reaktion auf eine einzelne BSM nicht von dem Zustand II in den Zustand I über, wie es möglich war, als die BSM sowohl das NMTSR **318B** als auch das CFR **320B** darstellte. Um von dem Zustand II in den Zustand I überzugehen, wenn die BSM nur das NMTSR **318B** darstellt, muß der Flußsteuerungsempfänger wenigstens zwei BSMs empfangen. Durch die erste BSM können die Register von dem Zustand II in den Zustand III übergehen, und durch die zweite BSM können die Register von dem Zustand III in den Zustand I übergehen. Die Register in dem Flußsteuerungsempfänger **304A** können jedoch von dem Zustand II in den Zustand I übergehen, falls ein Zeitüberschreitungereignis auftritt. Das Zeitüberschrei-

tungsergebnis ist oben unter Bezugnahme auf [Fig. 4](#) beschrieben. Die Operation des Flußsteuerungsprotokolls und -verfahrens ist unten unter Bezugnahme auf die [Fig. 6–Fig. 9](#) eingehender beschrieben.

**[0049]** [Fig. 6](#) ist ein Flußdiagramm des Flußsteuerungsprotokolls zum Senden eines Rahmens von dem ersten Router **104A** zu dem zweiten Router **104B** gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung. Der Arbiter 2 **204A** bestimmt bei **602**, ob irgendwelche Daten zu Port 6 **202B** des zweiten Routers **104B** über den Port 2 **202A** des ersten Routers **104A** zu senden sind. Nachdem bei **602** bestimmt ist, daß Daten zu senden sind, prüft der Arbiter 2 **204A** den Rahmenkopf, um die Priorität des Rahmens zu bestimmen. In der bevorzugten Ausführungsform hat jeder Rahmen eine von drei Prioritäten, z. B. eine niedrige, eine mittlere oder eine hohe Priorität. Der Flußsteuerungsempfänger **304A** in dem ersten Router **104A** bestimmt bei **606** die Anzahl von leeren Puffern in dem Pufferarray **322B** von Port 6. Bei dem Protokollbeispiel mit einer Priorität selektiert der Flußsteuerungsempfänger bei **608**, falls nur ein Puffer leer ist, diesen Puffer als Empfangspuffer und sendet die Adresse für den Empfangspuffer zu der Modifizierungseinheit **326A**, wie oben beschrieben. Der Flußsteuerungsempfänger **304A** bestimmt bei **610**, welche Rahmenpriorität erforderlich ist, um einen Rahmen von dem ersten Router **104A** zu dem zweiten Router **104B** zu senden. Falls zum Beispiel wenigstens ein Puffer in dem Pufferarray von Port 6 leer ist, sendet der Flußsteuerungsempfänger **304A** ein Signal zu dem Arbiter 2 auf einer Signalleitung mit hoher Priorität, das angibt, daß ein Rahmen mit hoher Priorität zu dem Port 6 **202B** gesendet werden kann. Falls wenigstens zwei Puffer in dem Pufferarray von Port 6 leer sind, sendet der Flußsteuerungsempfänger **304A** ein Signal zu dem Arbiter 2 auf der Signalleitung mit hoher Priorität und einer Signalleitung mit mittlerer Priorität, das angibt, daß ein Rahmen mit hoher Priorität oder ein Rahmen mit mittlerer Priorität zu dem Port 6 **202B** gesendet werden kann. Falls wenigstens drei Puffer in dem Pufferarray von Port 6 leer sind, sendet der Flußsteuerungsempfänger **304A** ein Signal zu dem Arbiter 2 auf der Signalleitung mit hoher Priorität, der Signalleitung mit mittlerer Priorität und einer Signalleitung mit niedriger Priorität, das angibt, daß ein Rahmen mit hoher Priorität, ein Rahmen mit mittlerer Priorität oder ein Rahmen mit niedriger Priorität zu dem Port 6 **202B** gesendet werden kann.

**[0050]** Der Arbiter 2 vergleicht die Rahmenpriorität in dem Rahmenkopf mit dem Wert des Signals, das auf den zugeordneten Prioritätssignalleitungen empfangen wurde, um bei **614** zu bestimmen, ob der Rahmen eine ausreichende Priorität hat, um gesendet zu werden. Falls die Prioritätssignalleitung angibt, daß der Rahmen nicht gesendet werden kann, bleibt

der Rahmen in dem Pufferarray **322A** von Port 3 bis zu solch einer Zeit, daß der Rahmen gesendet werden kann. Falls die Prioritätssignalleitung angibt, daß der Rahmen gesendet werden kann, wird der Rahmen bei **616** zu Port 6 **202B** des zweiten Routers **104B** auf die oben beschriebene Weise gesendet.

**[0051]** Der zweite Router **104B** empfängt den Rahmen bei **618**. Nachdem der Datensynchronisierer **324B** bestimmt, daß es ein Datenrahmen ist, speichert der Datensynchronisierer **324B** den Rahmen bei **620** in dem Empfangspuffer. Der Datensynchronisierer sendet die Daten zu dem Pufferarray **322B** von Port 6. Um zu bestimmen, welcher Puffer in dem Pufferarray **322B** von Port 6 der Empfangspuffer ist, kann das Pufferarray **322B** von Port 6 die Pufferadresse in dem Rahmenkopf lesen und dann den Rahmen mit dem Puffer multiplexieren, dessen Adresse in dem Rahmenkopf ist. Bei einer anderen Technik selektiert das Pufferarray **322B** von Port 6 den Empfangspuffer, bevor der Rahmen empfangen wird. Wenn der Rahmen empfangen ist, wird er sofort zu dem vorbestimmten Empfangspuffer gesendet. Die Pufferadresse in dem Rahmenkopf ist eine Pufferkennung und wird bei **624** der eigentlichen Pufferadresse zugeordnet. Wenn die BSM erzeugt wird, werden die Bits in dem NMTSR **318B** und dem CFR **320B** der Pufferkennungsadresse des Puffers in dem Pufferarray **322B** von Port 6 anstelle der tatsächlichen Pufferadresse entsprechen. Falls der Flußsteuerungsempfänger **304A** zum Beispiel den Puffer I, der eine Adresse hat, die 001 gleich ist, als Empfangspuffer selektiert, wird diese Adresse in dem Rahmenkopf gespeichert. Wenn dieser Rahmen zu dem zweiten Router **104B** gesendet wird, sendet der Datensynchronisierer **324B** den Rahmen zu dem Pufferarray **322B** von Port 6. Das Pufferarray **322B** von Port 6 kann den Puffer 3 mit der Adresse 011 als Empfangspuffer selektiert haben. Die Daten in dem Rahmen werden in dem Puffer 3 gespeichert. Wenn der BSM-Generator **316B** jedoch das NMTSR **318B** und das CFR **320B** speichert, wird der Status des Puffers an der Bitposition gespeichert, die der Kennungsadresse, d. h. 001, entspricht. Wenn die BSM durch den Flußsteuerungsempfänger **304A** in dem ersten Router **104A** empfangen wird, entsprechen die Werte von NMTSR **318B** und die Werte von CFR **320B** in der BSM daher dem Kennungswert, der durch den BSM-Empfänger **306A** erzeugt wird.

**[0052]** In der bevorzugten Ausführungsform wird nicht der gesamte Rahmen in dem Empfangspuffer gespeichert, es sei denn, daß der Rahmen nicht zu dem dritten Router **104E** weitergeleitet werden kann. Diese Virtual-Cut-Through-Routing-Technik wird dadurch erreicht, daß alle Routeninformationen in dem Rahmenkopf vorhanden sind. Deshalb ist, sobald der zweite Router **104B** den Rahmenkopf empfängt, der Ausgangsport in dem zweiten Router **104B** bekannt, der mit dem dritten Router **104E** verbunden ist, z. B.

Port 2, und der zweite Router **104B** kann mit dem Senden des Rahmenkopfes zu dem dritten Router **104E** beginnen, bevor der zweite Router **104B** den gesamten Rahmenkörper empfängt.

**[0053]** Nachdem der Empfangspuffer einen Abschnitt des Rahmens empfängt, sendet der Flußsteuerungssender **305B** in dem zweiten Router **104B** den Status des Empfangspuffers zu dem BSM-Generator **316B**. Falls der Empfangspuffer leer war, als die vorherige BSM erzeugt wurde, und Daten am Empfangspuffer empfangen werden, wird dann das zugeordnete Bit in dem NMTSR **318B** bei **626** gleich einer logischen Eins gesetzt. Das Bit in dem NMTSR **318B** wird für wenigstens eine BSM gleich einer logischen Eins bleiben, wie oben beschrieben, um als Bestätigung zu dienen, daß der Rahmen durch den zweiten Router **104B** empfangen worden ist. Wenn der Empfangspuffer nicht leer ist, wird zusätzlich das CFR **320B** bei **628** auch auf eine logische Eins gesetzt. Falls der Empfangspuffer jedoch leer wird, bevor die BSM erzeugt wird, wird das CFR **320B** bei **630** auf eine logische Null zurückgesetzt, um anzugeben, daß der Empfangspuffer leer ist. Der Inhalt des NMTSR **318B** und möglicherweise des CFR **320B** ist, wie oben beschrieben, in der BSM enthalten. Eine eingehendere Erläuterung des Sendens und Empfangens von BSMs folgt unten unter Bezugnahme auf die [Fig. 7–Fig. 9](#).

**[0054]** [Fig. 7](#) ist ein Flußdiagramm einer Pufferstatusmeldungserzeugungs- und -sendetechnik gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung. Der Flußsteuerungssender **305B** sendet eine BSM immer dann, wenn gerade keine Datenrahmen von dem zweiten Router **104B** zu dem ersten Router **104A** gesendet werden, d. h., wenn Rahmen gerade in der Richtung gesendet werden, die zu der Beschreibung bei dem obigen Beispiel entgegengesetzt ist. Falls der Fluß von Daten in dieser entgegengesetzten Richtung stark ist, kann eine BSM erst dann gesendet werden, wenn alle Datenrahmen gesendet worden sind. Es gibt jedoch Situationen, bei denen es wichtig ist, daß der Flußsteuerungssender **305B** eine BSM sofort zu dem Flußsteuerungsempfänger **304A** sendet. Ein Beispiel für diese Situation ist, wenn ein Puffer in dem Pufferarray **322B** von Port 6 verfügbar wird. Es kann wichtig sein, daß der Flußsteuerungsempfänger **304A** eine BSM empfängt, die angibt, daß ein Puffer verfügbar geworden ist, weil zum Beispiel Rahmen vorhanden sein können, die in Pufferarrays **322** in dem ersten Router **104A** gespeichert sind und auf die Verfügbarkeit von genügend Puffern in dem Pufferarray **322B** von Port 6 warten, so daß der Rahmen gesendet werden kann. Falls die Informationen in dem Flußsteuerungsempfänger **304A** zum Beispiel angeben, daß nur zwei Puffer in dem Pufferarray **322B** von Port 6 verfügbar sind, können dann keine Rahmen mit einer niedrigen Priorität von dem ersten Router **104A** zu

dem Port 6 **202B** des zweiten Routers **104B** gesendet werden, falls das oben beschriebene Prioritätsprotokoll genutzt wird. Wenn ein dritter Puffer in dem Pufferarray **322B** von Port 6 verfügbar wird, können diese Informationen zu dem Flußsteuerungsempfänger **304A** in Form einer BSM gesendet werden. Um eine BSM sofort zu senden, sendet der Flußsteuerungssender ein Signal zu dem Arbitrer 6 **204B** in dem zweiten Router **104B**, das fordert, eine BSM so bald wie möglich zu senden. Der Arbitrer 6 **204B** erzeugt ein Signal, daß den MUX **328B** in dem zweiten Router **104B** steuert. Als Reaktion auf dieses Signal kann der MUX **328B** die BSM auf der Signalleitung **212** multiplexieren, nachdem zum Beispiel der gegenwärtige Senderahmen vollständig gesendet worden ist.

**[0055]** Falls es bei **702** Zeit zum Senden einer BSM ist, erzeugt dann der BSM-Generator **316B** bei **704** eine BSM. Der Inhalt der BSM kann, wie oben beschrieben, entweder das NMTSR **318B** oder sowohl das NMTSR **318B** als auch das CFR **320B** sein. Der Inhalt des NMTSR **318B** ist dem Inhalt des CFR **320B** zu der Zeit der vorherigen BSM gleich, mit der Ausnahme, daß dann, falls der Wert des CFR **320B** eine logische Null war und seit dem Senden der vorherigen BSM einige Daten in dem Puffer empfangen wurden, der Wert in dem NMTSR **318B**, der dem Puffer zugeordnet ist, gleich einer logischen Eins gesetzt wird, um anzugeben, daß der Rahmen empfangen wurde, wie oben beschrieben. Der Flußsteuerungssender **305B** sendet bei **706** die BSM zu dem MUX **328B**. Der MUX **328B** sendet bei **708** die BSM dann zu dem ersten Router **104A**. Nach dem Senden des BSM-Rahmens zu dem MUX **328B** bei **706** setzt der Flußsteuerungssender **305B** bei **710** das NMTSR **318B** dem CFR **320B** gleich.

**[0056]** [Fig. 8](#) ist ein Flußdiagramm einer Pufferstatusmeldungsempfangstechnik gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung. Der Datensynchronisierer **324A** empfängt die BSM, bestimmt bei **802**, daß die BSM eine Pufferstatusmeldung und kein Datenrahmen ist, und sendet die BSM zu dem Flußsteuerungsempfänger **304A**. Der Flußsteuerungsempfänger **304A** selektiert bei **804** jeden Puffer und führt die übrigen Schritte in [Fig. 8](#) für jeden Puffer parallel aus. Der BSM-Empfänger **306A** in dem Flußsteuerungsempfänger **304A** bestimmt bei **806**, ob das Bit, das jedem Puffer in dem SANRR **308A** zugeordnet ist, gleich einer logischen Eins ist. Falls SANRR **308A** gleich einer logischen Eins ist, bestimmt dann der BSM-Empfänger **306A** bei **808**, ob das Bit, das jedem Puffer in dem Abschnitt der BSM des NMTSR **318B** zugeordnet ist, gleich einer logischen Eins ist. Falls NMTSR = 1 ist, wird dann bei **810** das SANRR **308A** gleich einer logischen Null gesetzt, und der Prozeß geht bei Schritt **812** weiter. Falls der Wert im SANRR **308A** gleich einer logischen Null ist, oder falls der Wert des NMTSR **318B** gleich einer logischen Null ist, geht dann der Prozeß

bei Schritt 812 weiter. Bei Schritt 812 wird jedes Bit in dem RCFR 310A gleich dem zugeordneten Bit in dem CFR 320B gesetzt, falls die BSM sowohl das NMTSR 318B als auch das CFR 320B darstellt. Falls die BSM nur das NMTSR 318B darstellt, setzt der BSM-Empfänger 306A bei 812 das RCFR 310A gleich dem zugeordneten Bit in dem Abschnitt der BSM des NMTSR 318B. Der BSM-Empfänger 306A setzt dann bei 814 jedes Bit in dem BBR 312A gleich dem logischen ODER des zugeordneten Bits in dem SANRR 308A und dem RCFR 310A.

**[0057]** Fig. 9(a)–9(g) sind Beispiele für Registerwerte gemäß einer bevorzugten Ausführungsform der vorliegenden Erfindung. In Fig. 9 entsprechen die drei Register mit der Bezeichnung RCFR 310A, SANRR 308A und BBR 312A den Werten dieser Register in dem Flußsteuerungsempfänger 304A in dem ersten Router 104A, wenn Rahmen von dem ersten Router 104A zu dem zweiten Router 104B gesendet werden. Die zwei Register mit der Bezeichnung CFR 320B und NMTSR 318B entsprechen den Werten dieser Register in dem Flußsteuerungssender 305B in dem zweiten Router 104B, wenn Rahmen von dem ersten Router 104A zu dem zweiten Router 104B gesendet werden. Fig. 9(a) zeigt die Werte der Register, wenn die Router 104 initialisiert sind. Wie in den Zustandsdiagrammen in Fig. 4 und Fig. 5 angegeben, werden die Werte in dem RCFR 310A und BBR 312A gleich einer logischen Eins gesetzt, während die Werte in den übrigen Registern gleich einer logischen Null gesetzt werden. Der Flußsteuerungssender 305B erzeugt eine erste BSM und sendet sie zu dem Flußsteuerungsempfänger 304A, wie oben beschrieben. Die Werte der Register nach dem Empfang der ersten BSM sind in Fig. 9(b) gezeigt. Die erste BSM ist entweder BSM = 0,0 oder BSM = 0, in Abhängigkeit von der Größe der ersten BSM. In jeder Situation gehen die Register in dem Flußsteuerungsempfänger 304A von dem Zustand III in den Zustand I über. Der übrige Abschnitt dieses Beispiels schließt den ersten Puffer in dem Pufferarray 322B von Port 6 ein, oder den Puffer, der der Kennung 001 zugeordnet ist, wie oben beschrieben. Der erste Router 104A sendet einen Rahmen zu dem zweiten Router 104B. Der Status der Register nach der Rahmenübertragung ist in Fig. 9(c) gezeigt. In Fig. 9(c) ist das SANRR 308A gleich einer logischen Eins gesetzt, da der erste Router 104A einen Rahmen gesendet hat, aber kein Bestätigungssignal empfangen hat. Da das SANRR 308A auf eine logische Eins gesetzt ist, wird auch das BBR 312A gleich einer logischen Eins gesetzt.

**[0058]** Nachdem der Rahmen durch das Pufferarray 322B von Port 6 empfangen worden ist und der Flußsteuerungssender das NMTSR 318B und das RCFR 320B aktualisiert hat, sind die Werte der Registerwerte in Fig. 9(d) gezeigt. In Fig. 9(d) sind die Werte des SANRR 308A, RCFR 310A und BBR 312A nicht ver-

ändert worden, da keine BSM durch den Flußsteuerungsempfänger 304A empfangen wurde. Das erste Bit in dem NMTSR 318B ist jedoch gleich einer logischen Eins, da Daten durch das Pufferarray 322B von Port 6 empfangen wurden, seit die erste BSM gesendet wurde. Da der erste Puffer nicht leer ist, wird das erste Bit in dem CFR 320B auch gleich einer logischen Eins gesetzt. Bevor jedoch eine zweite BSM gesendet wird, wird der Rahmen zu dem dritten Router 104E gesendet, und der erste Puffer wird leer. Deshalb ist der Wert des CFR 320B, wenn die zweite BSM erzeugt wird, gleich einer logischen Null. Dies ist in Fig. 9(e) gezeigt. Wenn die Registerwerte so wie in Fig. 9(e) sind, erzeugt und sendet der Flußsteuerungssender 305B eine zweite BSM. Falls die zweite BSM sowohl das NMTSR 318B als auch das CFR 320B enthält, sind die Werte der Register in dem Flußsteuerungsempfänger 304A nach dem Empfang der zweiten BSM in Fig. 9(g) gezeigt. In Fig. 9(g) ist der Wert von allen Bits in dem SANRR 308A, RCFR 310A und BBR 312A gleich einer logischen Null. Die Werte der Register in dem Flußsteuerungssender 305B sind alle auch gleich einer logischen Null, da der Wert des CFR 320B in dem NMTSR 318B gespeichert ist. Dies ist zu den Registern in dem Flußsteuerungsempfänger 304A äquivalent, die in Fig. 4 von dem Zustand II in den Zustand I übergehen. Falls die zweite BSM nur das NMTSR 318B enthält, ist der Wert des Registers, nachdem der Flußsteuerungsempfänger 304A die zweite BSM empfängt, in Fig. 9(f) gezeigt. Dies ist zu den Registern in dem Flußsteuerungsempfänger 304A äquivalent, die in Fig. 5 von dem Zustand II in den Zustand III übergehen. Zu einer späteren Zeit wird der Flußsteuerungssender 305B eine dritte BSM erzeugen und senden, die nur die Werte des NMTSR 318B enthält. Nach dem Empfang der dritten BSM sind die Werte der Register so wie in Fig. 9(g) gezeigt. Dies ist zu den Registern in dem Flußsteuerungsempfänger 304A äquivalent, die in Fig. 5 von dem Zustand III in den Zustand I übergehen.

## Patentansprüche

1. Verfahren zum Steuern des Sendens von Daten von einem ersten Netzelement mit einem ersten Flag zu einem zweiten Netzelement mit einem Empfangspuffer (322) und einem zweiten Flag, in einem Computernetz, das die Schritte umfaßt:
  - (a) Senden von Daten von dem genannten Netzelement zu dem zweiten Netzelement, falls ein Wert des ersten Flags einem ersten Wert gleich ist, der angibt, daß der Empfangspuffer zum Empfangen von Daten verfügbar ist;
  - (b) Setzen des ersten Flags auf einen zweiten Wert, der angibt, daß der Puffer zum Empfangen von Daten nicht verfügbar ist;
  - (c) Empfangen der Daten in dem Empfangspuffer;
  - (d) Setzen des zweiten Flags auf einen Wert, falls der Empfangspuffer die Daten empfangen hat und der

Empfangspuffer zum Empfangen von Daten verfügbar ist;

(e) Erzeugen eines Steuersignals mit einem Wert, der dem Wert des zweiten Flags gleich ist;

(f) Senden des Steuersignals von dem zweiten Netzelement zu dem ersten Netzelement; und

(g) Setzen des ersten Flags gleich dem Wert des Steuersignals.

2. Verfahren nach Anspruch 1, bei dem dann, falls der Empfangspuffer die Daten empfangen hat und der Empfangspuffer (322) zum Empfangen von Daten verfügbar ist, das zweite Flag auf einen zweiten Wert gesetzt wird.

3. Verfahren nach Anspruch 1, bei dem dann, falls der Empfangspuffer die Daten empfangen hat und der Empfangspuffer (322) zum Empfangen von Daten verfügbar ist, das zweite Flag auf den ersten Wert gesetzt wird.

4. Verfahren nach Anspruch 1 oder 2, ferner mit: Initialisieren des ersten Flags auf einen Anfangswert als Antwort auf das Initialisieren des Computernetzes;

Modifizieren des Anfangswertes des ersten Flags in einen ersten Wert, falls der Empfangspuffer zum Empfangen von Daten verfügbar ist; und bei dem das Setzen des ersten Flags auf einen zweiten Wert, der angibt, daß der Puffer (322) zum Empfangen von Daten nicht verfügbar ist, ferner die Angabe umfaßt, daß der Empfangspuffer in einem vollen Zustand ist und zum Empfangen von Daten von einem ersten Netzelement nicht verfügbar ist, unabhängig davon, wann der Empfangspuffer die Daten tatsächlich empfängt.

5. Verfahren nach Anspruch 2 oder 3, bei dem Schritt (b) ferner die Schritte umfaßt:

Setzen eines ersten Registers gleich einem dritten Wert, der angibt, daß das erste Netzelement Daten zu dem zweiten Netzelement gesendet hat und kein Steuersignal empfangen hat, das angibt, daß die Daten durch das zweite Netzelement empfangen wurden; und

Setzen des ersten Flags auf den zweiten Wert, wenn das erste Register gleich dem dritten Wert ist.

6. Verfahren nach Anspruch (3), bei dem Schritt (d) ferner die Schritte umfaßt:

Setzen eines zweiten Registers auf einen vierten Wert, der angibt, daß das zweite Netzelement die Daten empfangen hat, die durch das erste Netzelement gesendet wurden;

Setzen eines dritten Registers auf einen fünften Wert, der angibt, daß der Puffer zum Empfangen von Daten verfügbar ist; und

Setzen des zweiten Flags auf den ersten Wert, falls das zweite Register gleich dem vierten Wert ist und das dritte Register gleich dem fünften Wert ist.

7. Verfahren nach Anspruch 6, ferner mit dem Schritt:

Setzen des dritten Registers auf einen sechsten Wert, der angibt, daß der Puffer zum Empfangen von Daten nicht verfügbar ist.

8. Verfahren nach Anspruch 7, ferner mit dem Schritt:

Setzen des zweiten Flags gleich dem zweiten Wert, falls das zweite Register gleich dem vierten Wert ist und das dritte Register gleich dem sechsten Wert ist.

9. Verfahren nach Anspruch 8, bei dem Schritt (g) ferner die Schritte umfaßt:

Setzen des ersten Registers auf einen achten Wert, der angibt, daß das zweite Netzelement die Daten empfing, die durch das erste Netzelement gesendet wurden, falls das zweite Flag gleich dem zweiten Wert ist;

Setzen eines vierten Registers auf einen siebten Wert, falls das zweite Flag gleich dem zweiten Wert ist.

10. Verfahren nach Anspruch 3, bei dem Schritt (f) ferner die Schritte umfaßt:

Senden des Steuersignals zu einem Eingang eines Multiplexers (328) in dem zweiten Netzelement; Erzeugen eines ersten Signals mit einem dritten Wert, wenn gerade keine Daten von dem zweiten Netzelement zu dem ersten Netzelement gesendet werden oder das Steuersignal eine höhere Priorität als Daten hat, die gerade von dem zweiten Netz zu dem ersten Netz gesendet werden;

Senden des Steuersignals von dem zweiten Netzelement zu dem ersten Netzelement, falls das erste Signal den dritten Wert hat; und

Senden von Daten von dem zweiten Netzelement zu dem ersten Netzelement, falls das erste Signal nicht den dritten Wert hat.

11. Verfahren nach Anspruch 4, bei dem der Dialog zwischen den ersten und zweiten Flags über das Steuersignal asynchron erfolgt.

12. Verfahren nach Anspruch 4 oder 11, bei dem dann, falls der Empfangspuffer die Daten empfangen hat und der Empfangspuffer zum Empfangen von Daten verfügbar ist, das zweite Flag auf einen dritten Wert gesetzt wird; und ferner mit dem Setzen des zweiten Flags auf einen vierten Wert, falls der Empfangspuffer die Daten nicht empfangen hat oder der Empfangspuffer zum Empfangen von Daten nicht verfügbar ist.

13. Verfahren nach Anspruch 12, ferner mit:

Senden des Steuersignals zu einem Eingang eines Multiplexers (328) in dem zweiten Netzelement; Senden des Steuersignals zu dem ersten Netzelement, wenn gerade keine Daten von dem zweiten Netzelement zu dem ersten Netzelement gesendet

werden oder das Steuersignal eine höhere Priorität als zweite Daten hat, die gerade von dem zweiten Netz zu dem ersten Netz gesendet werden.

14. Verfahren nach Anspruch 13, ferner mit: Senden der zweiten Daten von dem zweiten Netzelement zu dem ersten Netzelement, falls das Steuersignal keine höhere Priorität als die genannten Daten hat.

15. Verfahren nach irgendeinem der vorhergehenden Ansprüche, ferner mit dem Schritt:  
(h) Wiederholen der Schritte (a)–(g).

16. Verfahren nach irgendeinem der vorhergehenden Ansprüche, bei dem das Computernetz ein Ursprungsknoten-Routing-Netz ist, das alle Netz-Routing-Informationen hat, die in einem Sendeelement bestimmt werden.

17. Verfahren nach irgendeinem der vorhergehenden Ansprüche, bei dem das erste Netzelement und das zweite Netzelement Router sind (104).

18. System zum Optimieren des Sendens von Daten in einem Computernetz, welches System umfaßt:

ein erstes Netzelement mit einem ersten Flag; und ein zweites Netzelement, das mit dem ersten Netzelement gekoppelt ist, mit einem Empfangspuffer und einem zweiten Flag;

bei dem das erste Netzelement enthält:

eine erste Sendeeinheit zum Senden von Daten von dem genannten Netzelement, falls ein Wert des ersten Flags gleich einem ersten Wert ist, der angibt, daß der Empfangspuffer zum Empfangen von Daten verfügbar ist; und

eine erste Pufferstatuseinheit, die mit der ersten Sendeeinheit gekoppelt ist, zum Setzen des ersten Flags auf einen zweiten Wert, falls jener Puffer zum Empfangen von Daten nicht verfügbar ist, oder auf einen ersten Wert, falls der Puffer zum Empfangen von Daten verfügbar ist; und

bei dem das zweite Netzelement enthält:

ein Empfangselement, das mit der ersten Sendeeinheit gekoppelt ist, zum Empfangen der Daten, die von dem ersten Netzelement gesendet werden;

eine zweite Pufferstatuseinheit, die mit dem Empfangspuffer (322) gekoppelt ist und mit dem Empfangselement gekoppelt ist, zum Setzen des zweiten Flags auf den ersten Wert, falls der Empfangspuffer die Daten empfangen hat und der Empfangspuffer zum Empfangen von Daten verfügbar ist;

einen Steuersignalgenerator, der mit der zweiten Pufferstatuseinheit gekoppelt ist, zum Erzeugen eines Steuersignals, das einen Wert hat, der dem Wert des zweiten Flags gleich ist; und

eine zweite Sendeeinheit, die mit dem Steuersignalgenerator gekoppelt ist, zum Senden des Steuersignals von dem zweiten Netzelement zu dem ersten

Netzelement, bei dem die erste Pufferstatuseinheit das erste Flag auf den Wert des Steuersignals setzt.

19. System nach Anspruch 18, bei dem das Computernetz ein Ursprungsknoten-Routing-Netz ist, das alle Netz-Routing-Informationen hat, die in einem Datensendeelement bestimmt werden.

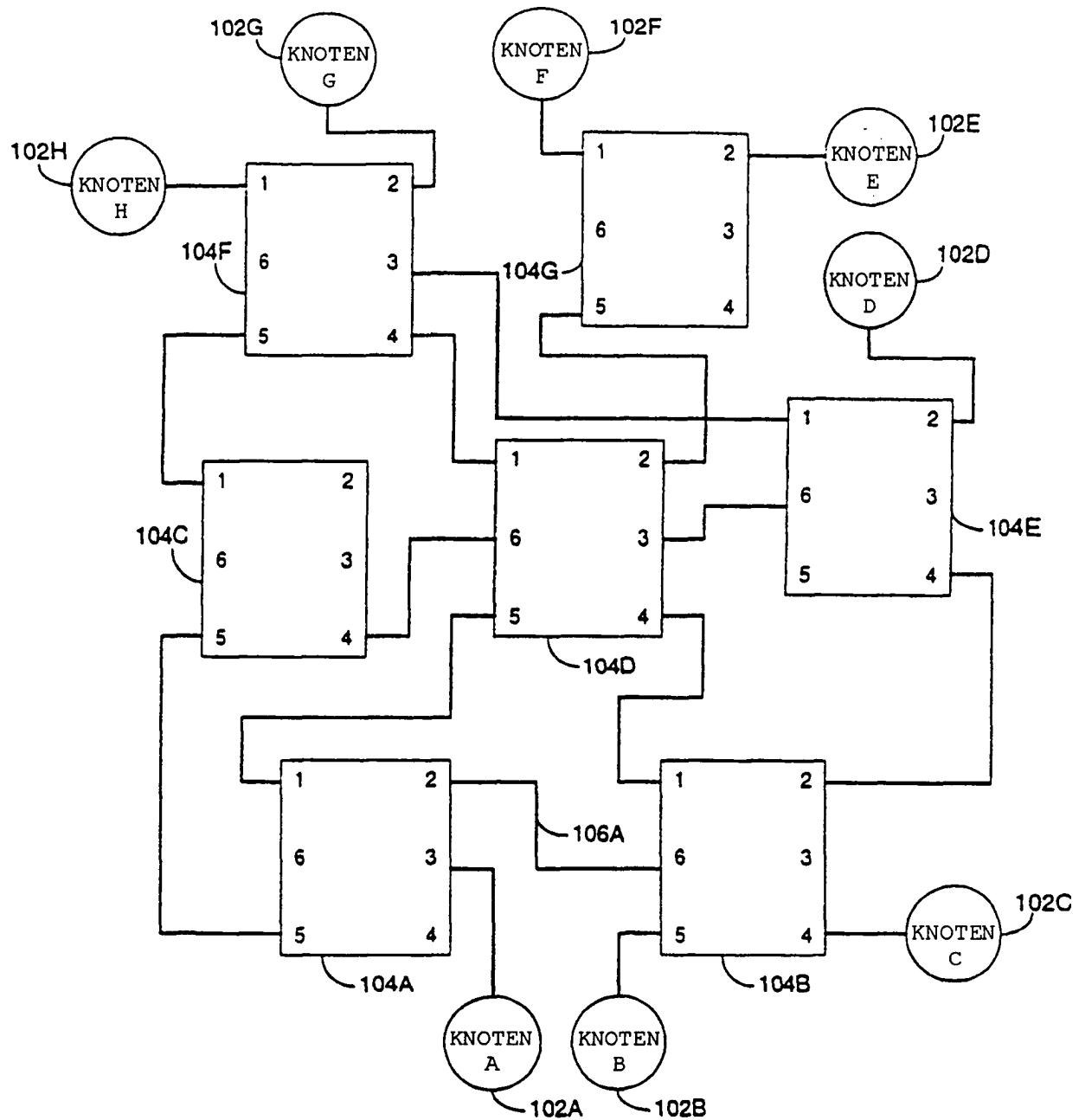
20. System nach Anspruch 18, bei dem das erste Netzelement und das zweite Netzelement Router (104) sind.

21. System nach Anspruch 18, bei dem die erste Pufferstatuseinheit, die mit der ersten Sendeeinheit gekoppelt ist, angibt, daß der Empfangspuffer (322) in einem vollen Zustand ist und zum Empfangen von Daten von dem ersten Netzelement nicht verfügbar ist, unabhängig davon, wann der Empfangspuffer die Daten tatsächlich empfängt, indem sie das erste Flag auf den zweiten Wert setzt.

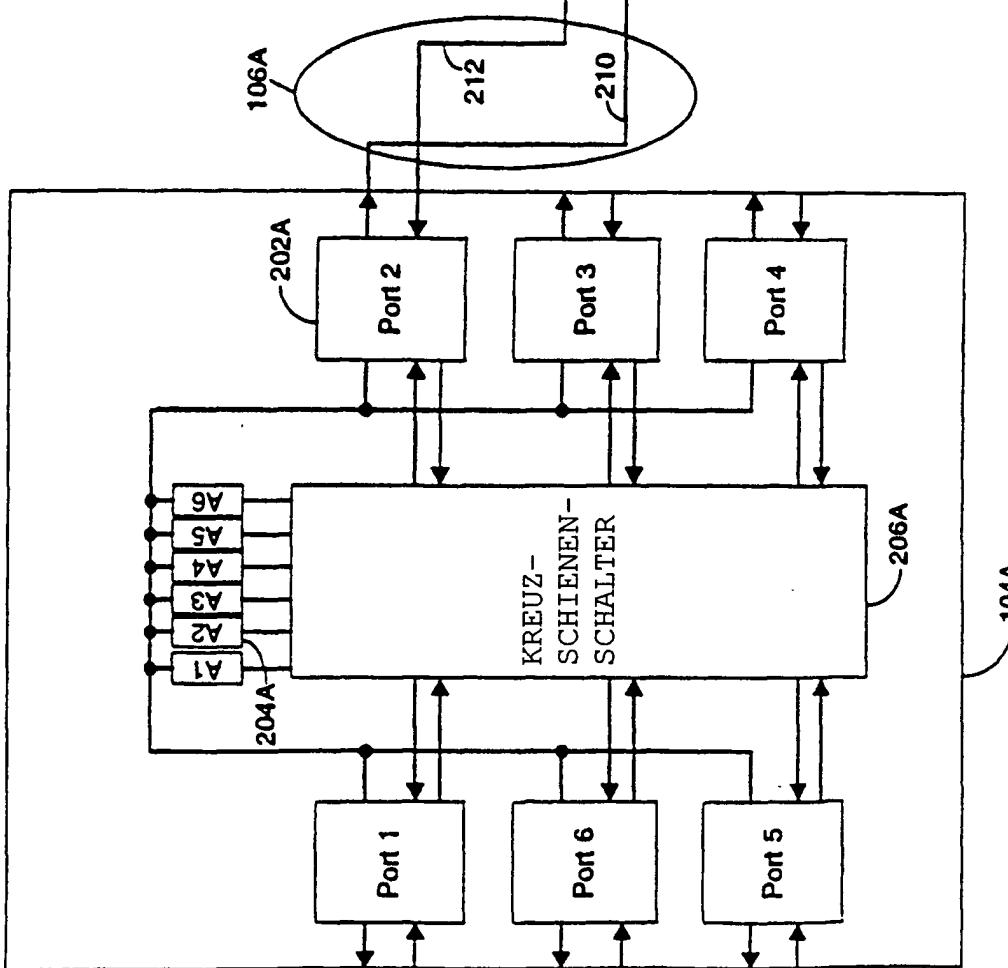
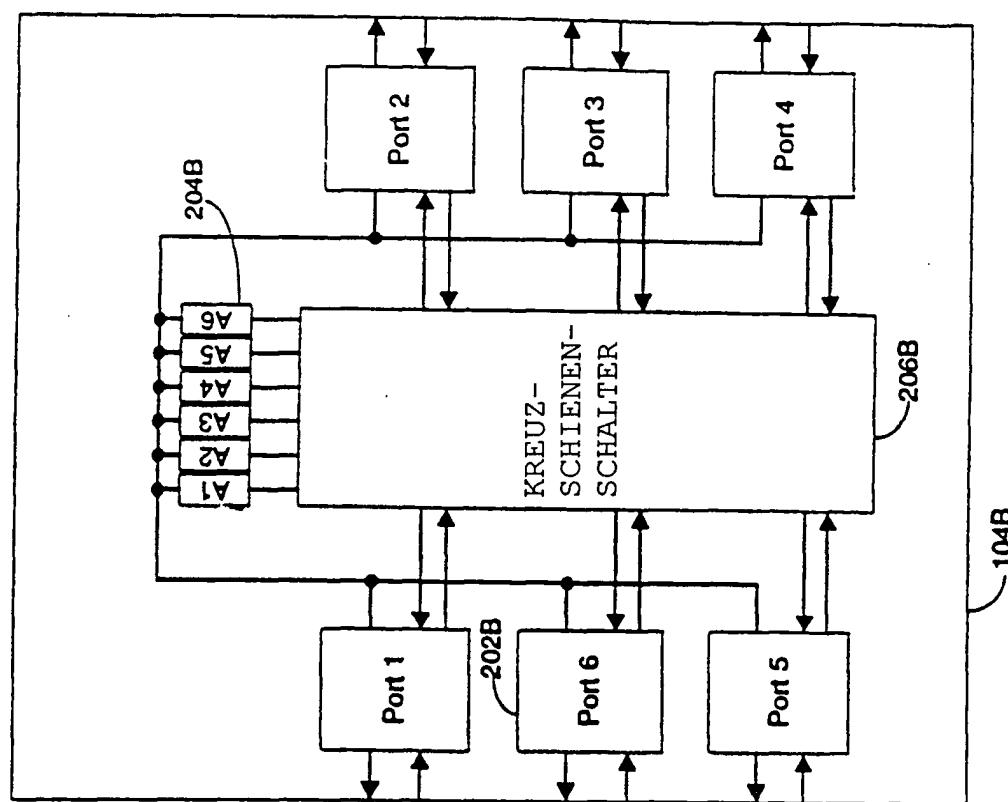
22. System nach Anspruch 18, bei dem der Dialog zwischen den ersten und zweiten Flags über das Steuersignal asynchron erfolgt.

Es folgen 9 Blatt Zeichnungen

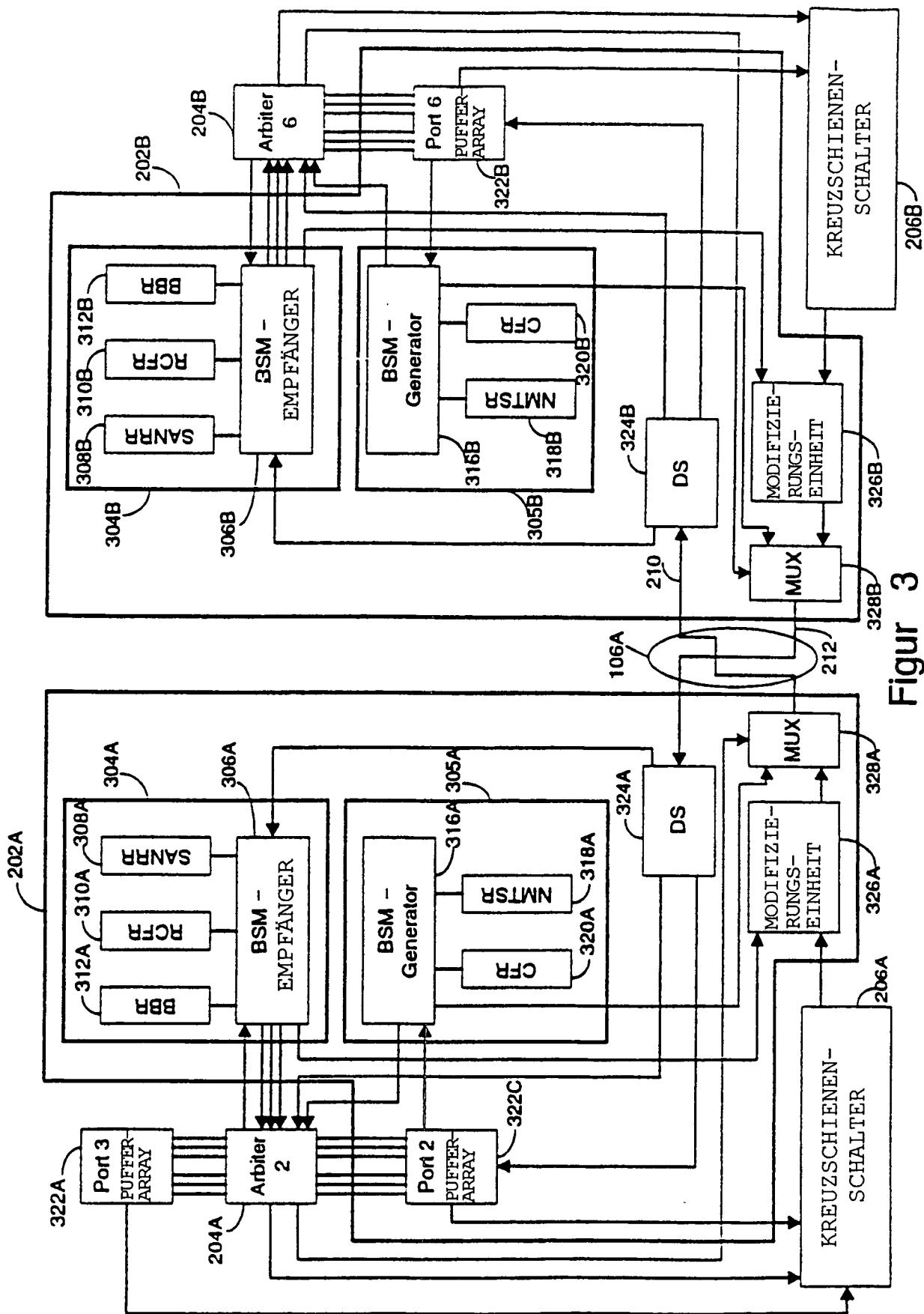
## Anhängende Zeichnungen



Figur 1

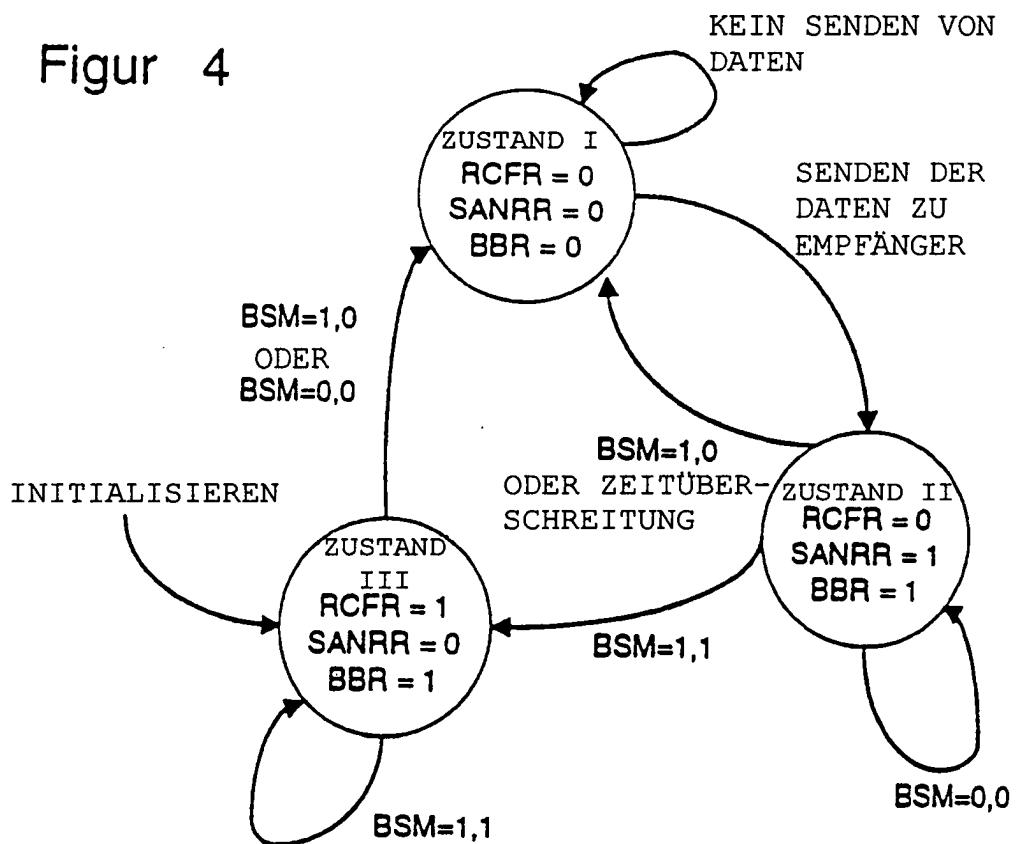


Figur 2

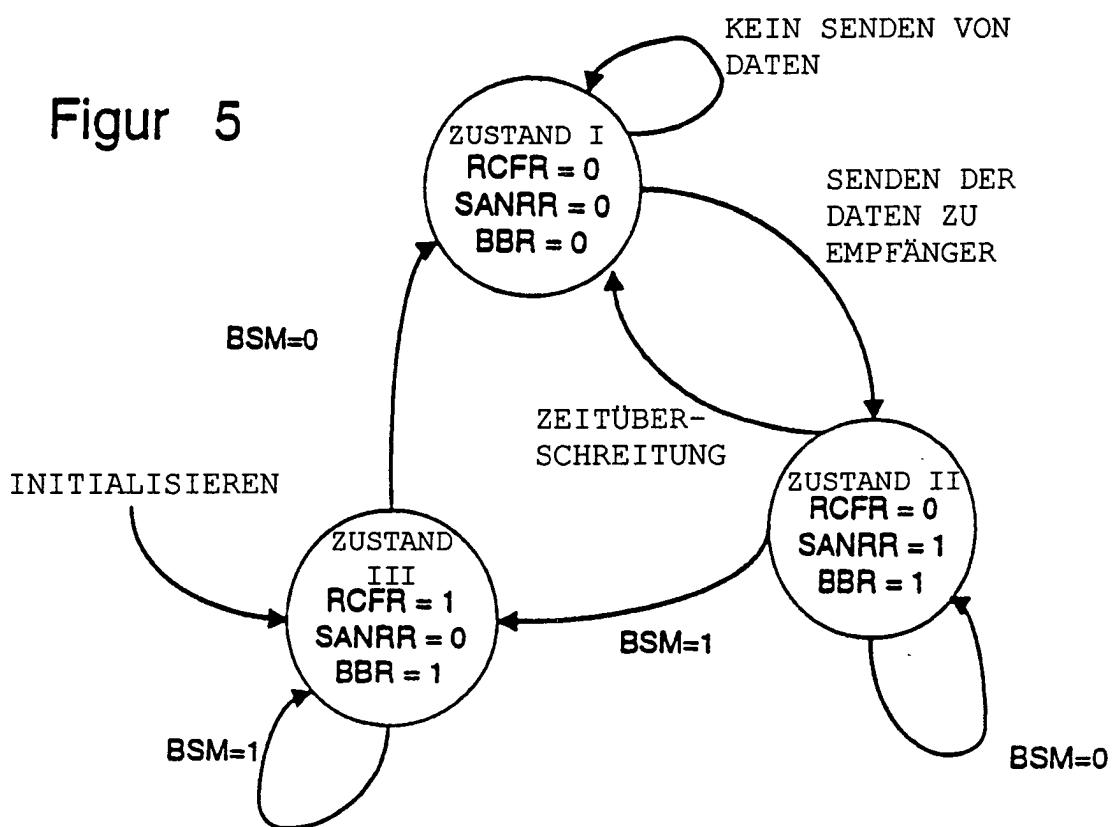


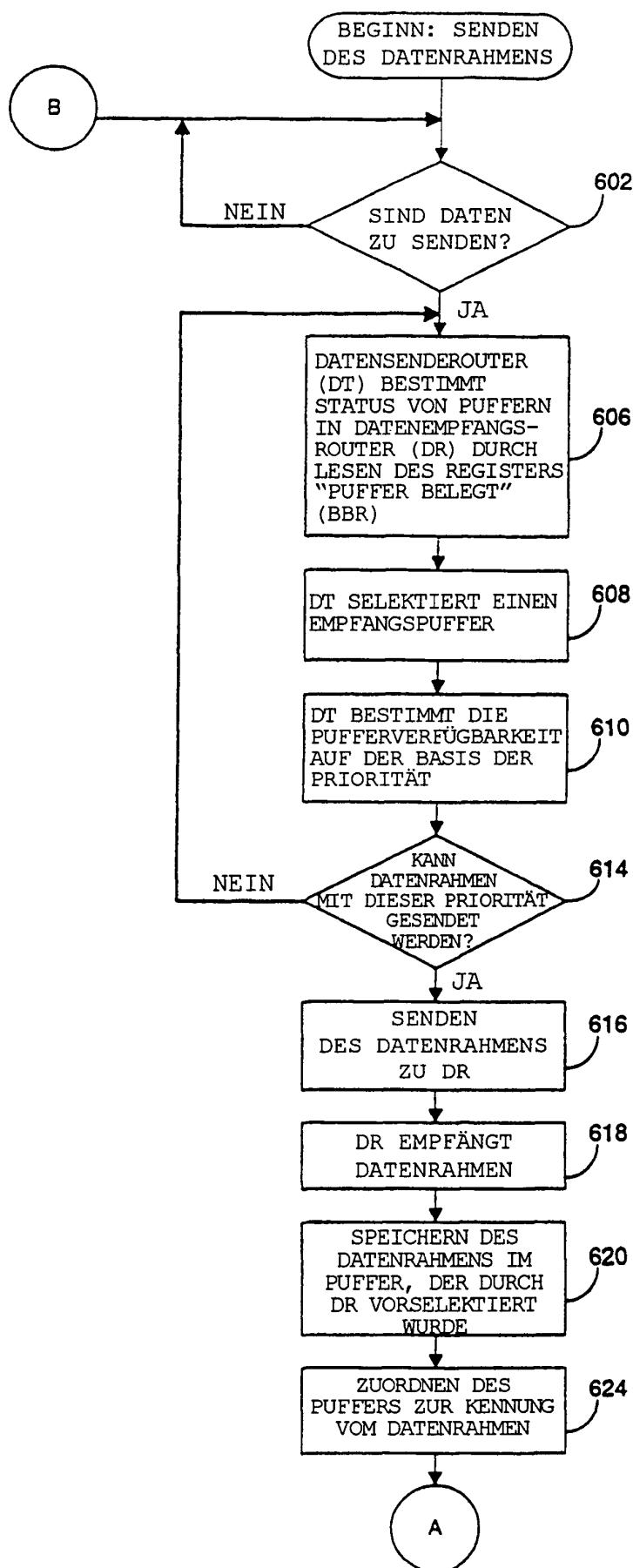
Figur 3

Figur 4

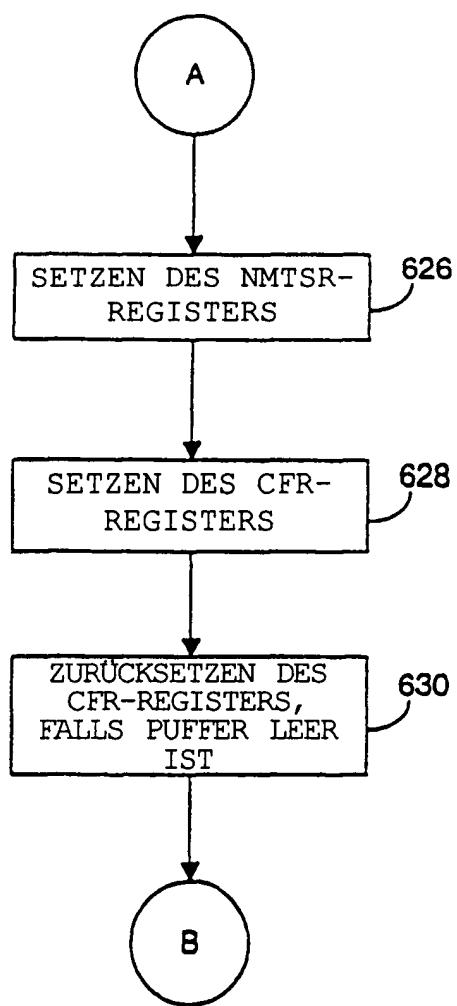


Figur 5

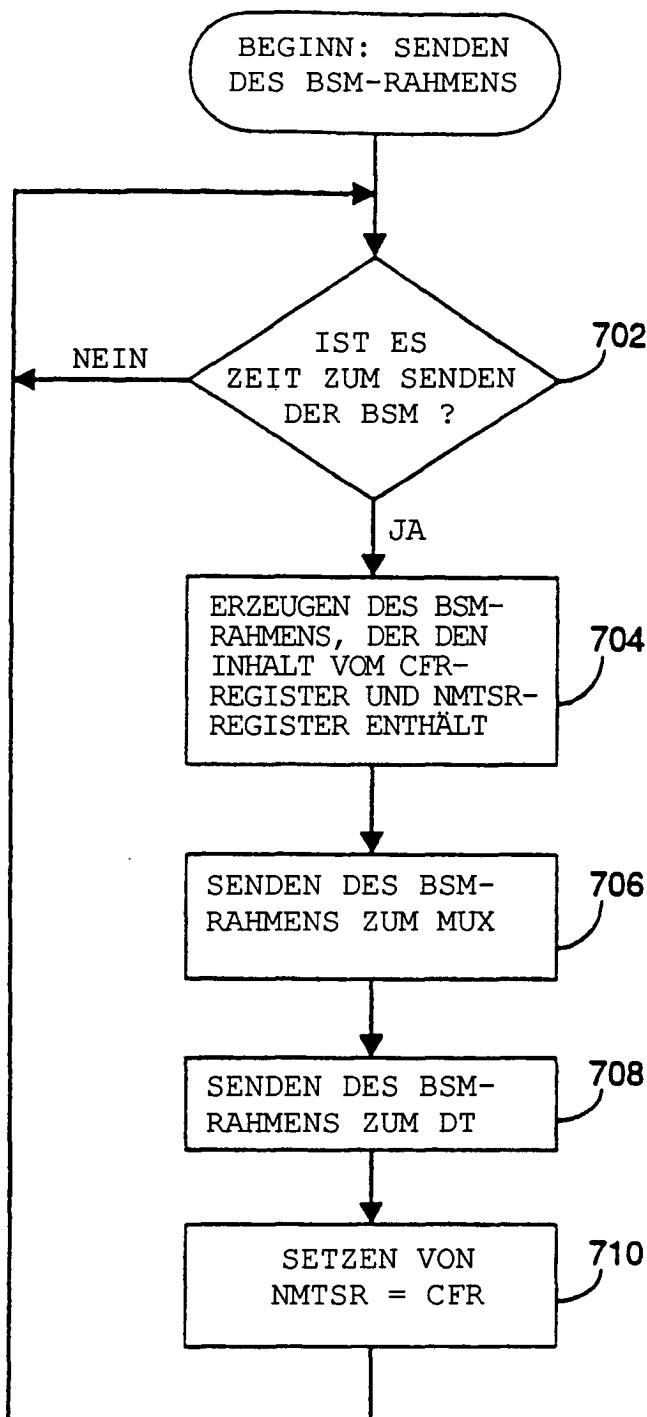




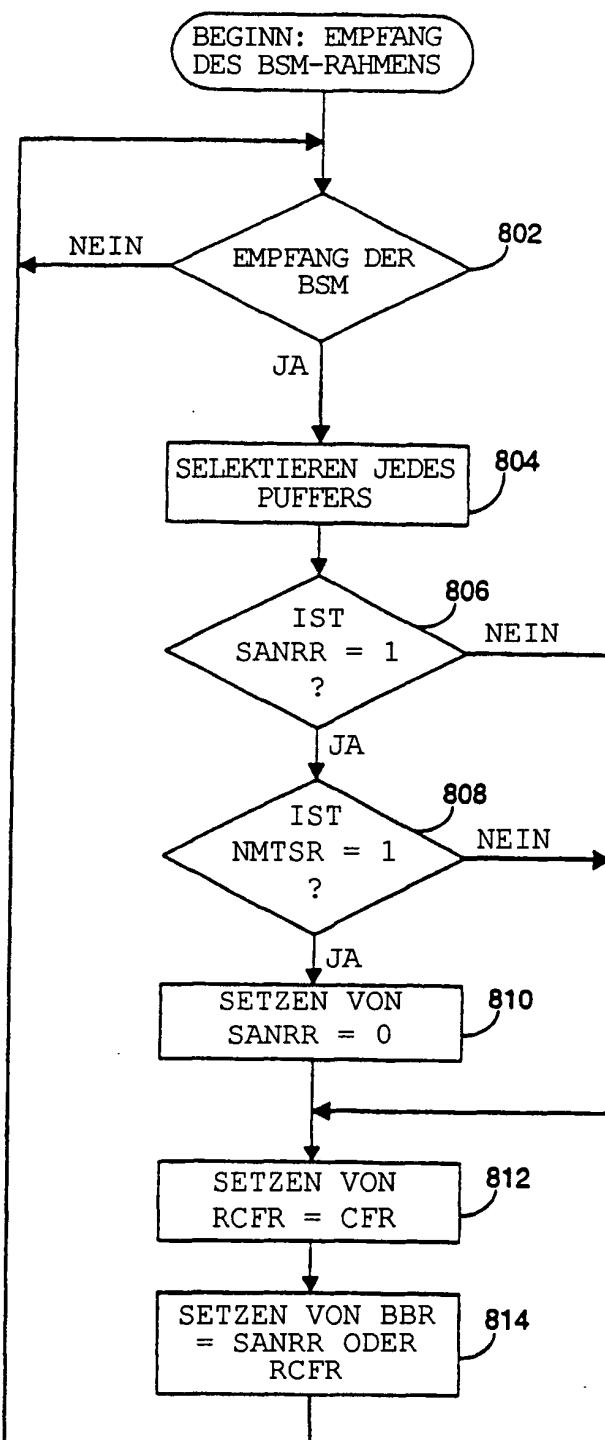
Figur 6(a)



Figur 6(b)



Figur 7



Figur 8

FLUSSSTEUERUNGSEMPFÄNGER  
304A IN ERSTEM ROUTER  
104A

FLUSSSTEUERUNGSSENDER  
305B IN ZWEITEM ROUTER  
104B

(a)	RCFR SANRR BBR	<table border="1"><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr></table>	1	1	1	1	1	1	0	0	0	0	0	0	1	1	1	1	1	1	<table border="1"><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	CFR NMTSR
1	1	1	1	1	1																													
0	0	0	0	0	0																													
1	1	1	1	1	1																													
0	0	0	0	0	0																													
0	0	0	0	0	0																													
(b)	RCFR SANRR BBR	<table border="1"><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	<table border="1"><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	CFR NMTSR
0	0	0	0	0	0																													
0	0	0	0	0	0																													
0	0	0	0	0	0																													
0	0	0	0	0	0																													
0	0	0	0	0	0																													
(c)	RCFR SANRR BBR	<table border="1"><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	<table border="1"><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	CFR NMTSR
0	0	0	0	0	0																													
1	0	0	0	0	0																													
1	0	0	0	0	0																													
0	0	0	0	0	0																													
0	0	0	0	0	0																													
(d)	RCFR SANRR BBR	<table border="1"><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	<table border="1"><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	1	0	0	0	0	0	1	0	0	0	0	0	CFR NMTSR
0	0	0	0	0	0																													
1	0	0	0	0	0																													
1	0	0	0	0	0																													
1	0	0	0	0	0																													
1	0	0	0	0	0																													
(e)	RCFR SANRR BBR	<table border="1"><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	1	0	0	0	0	0	1	0	0	0	0	0	1	0	0	0	0	0	<table border="1"><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	1	0	0	0	0	0	CFR NMTSR
1	0	0	0	0	0																													
1	0	0	0	0	0																													
1	0	0	0	0	0																													
0	0	0	0	0	0																													
1	0	0	0	0	0																													
(f)	RCFR SANRR BBR	<table border="1"><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	<table border="1"><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	CFR NMTSR
1	0	0	0	0	0																													
0	0	0	0	0	0																													
1	0	0	0	0	0																													
0	0	0	0	0	0																													
0	0	0	0	0	0																													
(g)	RCFR SANRR BBR	<table border="1"><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	<table border="1"><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table>	0	0	0	0	0	0	0	0	0	0	0	0	CFR NMTSR
0	0	0	0	0	0																													
0	0	0	0	0	0																													
0	0	0	0	0	0																													
0	0	0	0	0	0																													
0	0	0	0	0	0																													

Figur 9