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

(10) **DE 602 21 462 T2** 2008.04.17

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 1 267 287 B1**
(21) Deutsches Aktenzeichen: **602 21 462.9**
(96) Europäisches Aktenzeichen: **02 253 932.4**
(96) Europäischer Anmeldetag: **06.06.2002**
(97) Erstveröffentlichung durch das EPA: **18.12.2002**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **01.08.2007**
(47) Veröffentlichungstag im Patentblatt: **17.04.2008**

(51) Int Cl.[8]: ***G06F 17/50*** (2006.01)

(30) Unionspriorität:
2001176375 11.06.2001 JP

(84) Benannte Vertragsstaaten:
DE, FR, GB

(73) Patentinhaber:
Sharp K.K., Osaka, JP

(72) Erfinder:
Ohnishi, Mitsuhisa, Tenri-shi, Nara, JP

(74) Vertreter:
Müller - Hoffmann & Partner Patentanwälte, 81667 München

(54) Bezeichnung: **Vorrichtung und Verfahren zur High-Level-Synthese, Verfahren zur Produktion von logischen Schaltungen unter Verwendung des Verfahrens zur High-Level-Synthese,und Aufzeichnungsmedium**

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

HINTERGRUND DER ERFINDUNG

1. GEBIET DER ERFINDUNG

[0001] Die Erfindung betrifft eine High-Level-Synthesevorrichtung zum automatischen Erzeugen einer Logikschaltung für eine integrierte Halbleiterschaltung (LSI), die durch eine Verhaltensbeschreibung (Designdaten) repräsentiert ist, in der Verarbeitungsverhaltensweisen der Logikschaltung beschrieben sind. Die Erfindung betrifft auch ein High-Level-Syntheseverfahren unter Verwendung der High-Level-Synthesevorrichtung, ein Verfahren zum Herstellen einer Logikschaltung unter Verwendung des High-Level-Syntheseverfahrens sowie einen Computer-lesbaren Aufzeichnungsträger, der ein Steuerungsprogramm zum Ausführen des High-Level-Syntheseverfahrens speichert.

2. BESCHREIBUNG DER EINSCHLÄGIGEN TECHNIK

[0002] Jüngere Mikrobearbeitungstechnologien haben größere System-LSIs ermöglicht. Es wird nach einer Entwicklungsumgebung gesucht, in der derartige System-LSIs effizient konzipiert und getestet werden können.

[0003] In den 1990-ern wurde ein Logiksynthesewerkzeug bis zur praktischen Verwendung entwickelt. Darauffolgend wurde ein Verhaltenssynthesewerkzeug zum Synthetisieren einer Beschreibung auf Registertransferlevel (nachfolgend als RT-Level bezeichnet) auf Grundlage einer Verhaltensbeschreibung, gemäß der nur Verhaltensweisen ausschließlich Information zu einer Hardwarestruktur beschrieben sind, in Gebrauch gebracht. Das Verhaltenssynthesewerkzeug erzeugt LSI-Designs, die manuell erzeugten vergleichbar sind, innerhalb einer kürzeren Zeitspanne.

[0004] Wenn ein derartiges Verhaltenssynthesewerkzeug verwendet wird, kann ein Designer seine Anstrengungen auf das Konzipieren eines Algorithmus, der eine wesentliche Verhaltensweise eines LSI bestimmt, konzentrieren. Ein derartiges Konzipieren eines Algorithmus stützt sich stark auf manuelle Arbeit. Im Ergebnis kann die Qualität einer Schaltung verbessert werden.

[0005] In frühen Stadien des Konzipierens eines großen digitalen LSI, wie eines System-LSI, wird als erstes ein Algorithmus eines Gesamtsystems untersucht und getestet (dieser Prozess wird als "Algorithmusdesign" bezeichnet). Hierbei wird eine Softwarebeschreibungssprache, wie eine Programmiersprache (beispielsweise die "Sprache C") dazu verwendet, einen Algorithmus auf einer Workstation oder einem Personal Computer zu konzipieren und zu testen.

[0006] Anschließend werden individuelle Prozesse, wie sie innerhalb eines Systems benötigt werden, durch eine Hardwarebeschreibungssprache mit Verhaltensbeschreibungen, die zu testen sind, angegeben. Daher wird ein zuvor durch eine Softwarebeschreibungssprache angegebener Algorithmus erneut durch eine Hardwarebeschreibungssprache mit einer Verhaltensbeschreibung angegeben.

[0007] Demgemäß ist herkömmlicherweise ein Verfahren (High-Level-Syntheseverfahren) zum Synthetisieren einer Schaltung auf Grundlage eines Algorithmus eines Gesamtsystems oder einer Verhaltensbeschreibung unter Verwendung der Sprache C vorgeschlagen. Eine derartige herkömmliche Technik ist beispielsweise in der japanischen Patentoffenlegungsveröffentlichung Nr. 10-116802 mit dem Titel "Verfahren zum Konzipieren eines integrierten Schaltkreises sowie durch das Verfahren konzipierter integrierter Schaltkreis" offenbart.

[0008] Derzeit wird eine Sprache mit hohem Abstraktionsgrad, wie die "Sprache C" dazu verwendet, das Verhalten von Hardware, die eine Anwendung, wie eine Audio- oder Videoverarbeitung, realisiert, zu beschreiben und eine Hardwareschaltung zu synthetisieren (High-Level-Synthese).

[0009] Als erstes wird eine Verhaltensbeschreibungssprache beschrieben. Danach wird die Sprache C in der folgenden Beschreibung für den Zweck einer Verhaltensbeschreibung erweitert. Genauer gesagt, beinhaltet die erweiterte Sprache C Parallelsätze zum expliziten Beschreiben paralleler Operationen, Befehle für Datenkommunikation zwischen den parallelen Operationen sowie Kommunikationskanäle.

[0010] Ein Beispiel einer derartigen Sprache ist die in "Bach: Environment for SLI Design with C Language", The 11th Workshop an Circuits and Systems in Karuisawa, April 20-21, 1998, and "Hardware Complier Bach", TECHNICAL REPORT OF IEICE CRSY97-87(1997-10) offenbarte Sprache Bach C. Die Fig. 15 zeigt ein Beispiel der Sprache C zur Hardwarebeschreibung. Bei diesem Beispiel ist das folgende Verhalten beschrieben.

[0011] Wie es in der Fig. 15 dargestellt ist, ist die dritte Zeile der Verhaltensbeschreibung die Deklaration eines synchronen Kommunikationskanals ch vom Typ int.

[0012] "Par" in der sechsten Zeile ist ein Parallelsatz, der explizit Paralleloperationen angibt. Hierbei gibt der Satz an, dass dabei zwei Blöcke parallel einer Operation unterzogen werden.

[0013] Die siebte Zeile beschreibt eine erste Threadoperation, die angibt, dass ein Kommunikationsdatenbefehl "send" dazu verwendet wird, den Wert 10 an den Kommunikationskanal ch zu senden.

[0014] Die achte Zeile gibt die nächste Threadoperation an, die anzeigt, dass ein Kommunikationsdatenbefehl "receive" dazu verwendet wird, Daten vom Kommunikationskanal ch zu empfangen, wobei die empfangenen Daten in eine Variable x eingesetzt werden.

[0015] Die zehnte Zeile zeigt an, dass nach dem Ausführen des Parallelsatzes in den Zeilen 6 bis 9 der Inhalt der Variablen x als Dezimalzahl im Integertyp an "stdout" ausgegeben wird.

[0016] Als nächstes werden die Prozesse einer High-Level-Synthese auf Grundlage einer beispielhaften High-Level-Synthese erläutert, wie sie durch die japanische Patentoffenlegungsveröffentlichung Nr. 10-116302 mit dem Titel "Verfahren zum Konzipieren eines integrierten Schaltkreises sowie durch das Verfahren konzipierter integrierter Schaltkreis" vorgeschlagen ist. Der Ablauf der High-Level-Synthese ist grob in vier Stadien, nämlich (N1) bis (N4) unterteilt. Die Erläuterung erfolgt unter Bezugnahme auf die Funktionsblöcke in der Fig. 16.
(N1) Es wird eine Verhaltensbeschreibung analysiert, die einen Algorithmus des Verarbeitungsverhaltens einer Schaltung beschreibt.
(N2) Das Verarbeitungsverhalten wird in Threads unterteilt, die parallel asynchron abgearbeitet werden.
(N3) Für jeden Thread werden die folgenden Prozesse (N3a) bis (N3f) ausgeführt.
(N3a) CDFG-Synthese
ein CDFG (control data flow graph) ist ein Diagramm, das Abhängigkeitsbeziehungen zwischen Rechenvorgängen hinsichtlich der Ausführungsreihenfolge repräsentiert. In einem CDFG sind Rechenvorgänge, Eingaben und Ausgaben durch Knoten repräsentiert, und Datenabhängigkeitsbeziehungen sind durch direkte Zweige repräsentiert.
(N3b) Zeitplanung
Eine einem als Schritt bezeichneten Takt entsprechende Zeit wird sukzessive jedem der Rechenvorgänge, der Eingaben und Ausgaben im CDFG zugeordnet.
(N3c) Zuordnung
Es werden Berechnungseinheiten, Register sowie Eingabe- und Ausgabepins, wie sie zur Ausführung des zeitlich geplanten CDFG erforderlich sind, erzeugt. Die Berechnungseinheiten werden den Rechenvorgängen im CDFG zugeordnet. Die Register werden Datenabhängigkeitszweigen über Taktgrenzen hinweg zugeordnet. Die Eingabe- und Ausgabepins werden den Eingaben und Ausgaben zugeordnet.
(N3d) Erzeugung eines Datenpfads
Es werden Datenpfade erzeugt, die jeweiligen Datenabhängigkeitsbeziehungszweigen im CDFG entsprechen.
(N3e) Erzeugung eines Steuerungsmechanismus
Es wird ein Steuerungsmechanismus zum Steuern der Berechnungseinheiten, Register und Multiplexer, wie sie während der Zuordnung und der Erzeugung der Datenpfade erzeugt werden, erzeugt.
(N3f) Erzeugung einer Schaltung auf RT-Level
Es wird eine Schaltungsbeschreibung der durch die oben beschriebenen Prozesse erzeugten Schaltung unter Verwendung einer Hardwarebeschreibungssprache, wie VHDL [VHSIC (Very High Speed Integrated Circuit) Hardware Description Language], und dergleichen, erzeugt.
(N4) Es werden Schaltungen auf RT-Level, eine jeweilige für die jeweiligen Threads (Teilschaltkreise) zu einer einzelnen Schaltung auf RT-Level integriert.

[0017] Als nächstes werden die Paralleloperationen und die Kommunikation bei der High-Level-Synthese erläutert.

[0018] Nachfolgend wird davon ausgegangen, dass bei Kommunikationsvorgängen unter Verwendung synchroner Kanäle (nachfolgend als Kommunikation über synchrone Kanäle bezeichnet) Daten übertragen werden, nachdem sowohl ein senderseitiger Thread (nachfolgend als Sendethread bezeichnet) als auch ein empfängerseitiger Thread (nachfolgend als Empfangsthread bezeichnet) im Zustand einer Kommunikationsbereitschaft vorliegen.

[0019] Als ein Verfahren zum Realisieren einer Kommunikation über synchrone Kanäle kann eine Schaltungskonfiguration verwendet werden, wie sie in der Fig. 17 dargestellt ist, die ein Quittierungssteuerungssignal verwendet. Bei einer derartigen Schaltungskonfiguration verfügt eine Sendethreadschaltung über die folgenden Ports. Ein (I) im Namen jedes Ports der Schaltung zeigt an, dass der Port ein Eingangsport ist, wohingegen (O) anzeigt, dass der Port ein Ausgangsport ist. "wtx" kennzeichnet einen Steuerungsleitungsport für ein Sendeanforderungssignal (oder ein Sendeabschlusssignal) an ein Empfängerende. "wrx" repräsentiert einen Steuerleitungspfad für ein Empfangsanforderungssignal (oder ein Empfangsabschlusssignal) von einem Empfängerende. "wdata" repräsentiert einen Datenleitungsport auf einem Senderende.

[0020] Andererseits verfügt die Empfangsthreadschaltung über die folgenden Ports. Ein (I) im Namen jedes Ports der Schaltung zeigt an, dass der Port ein Eingangsport ist, während (O) anzeigt, dass der Port ein Ausgangsport ist. "rrx" repräsentiert einen Steuerleitungsport für ein Empfangsanforderungssignal (oder ein Empfangsabschlusssignal) an einem Empfängerende. "rtx" repräsentiert einen Steuerleitungsport für ein Sendeanforderungssignal (oder ein Sendeabschlusssignal) von einem Empfängerende. "rda-

ta" repräsentiert einen Datenleitungsport auf einem Empfängerende.

[0021] Die Fig. 18 zeigt ein beispielhaftes Timingdiagramm für die Datenübertragung innerhalb dieser Schaltungskonfiguration. Es sei darauf hingewiesen, dass die Spannung jeder Steuerungsleitung im Ausgangszustand als Pegel "NIEDRIG" angenommen ist.

[0022] Es wird davon ausgegangen, dass als erstes ein "Sendebefehl" ausgeführt wird und danach ein "Empfangsbefehl" ausgeführt wird. Wie es in der Fig. 18 dargestellt ist, führt beispielsweise ein Sendethread (thread 1) einen "Sendebefehl" innerhalb eines Taktzyklus c1 aus, und ein Empfangsthread (thread 2) führt einen "Empfangsbefehl" innerhalb eines Taktzyklus c3 aus.

[0023] Der Sendethread gibt Daten d1 im Taktzyklus c1 an den Datenleitungsport "wdata" aus, und er sorgt dafür, dass die Spannung am Steuerleitungsport "wtx(O)" der Pegel "HOCH" ist. Ferner wartet der Sendethread, bis die Spannung am Steuerleitungsport "wrx(I)" auf "HOCH" geht. Wenn die Spannung am Steuerleitungsport "wrx(I)" auf "HOCH" geht, sorgt der Sendethread dafür, dass die Spannung am Steuerleitungsport "wtx(O)" auf "NIEDRIG" geht, und er beendet im nächsten Taktzyklus (c4) die Datenausgabe an den Datenleitungsport "wdata".

[0024] Andererseits sorgt der Empfangsthread dafür, dass die Spannung am Steuerleitungsport "rrx(O)" im Taktzyklus c3 den Pegel "HOCH" einnimmt. Hierbei nimmt der Empfangsthread, da die Spannung am Steuerleitungsport "rtx(I)" dem Pegel "HOCH" entspricht, auf die Daten d1 am Datenleitungsport "rdata" Bezug, und er sorgt dafür, dass die Spannung am Steuerleitungsport "rrx(O)" im nächsten Taktzyklus c4 auf "NIEDRIG" geht. So wird die Datenkommunikation beendet.

[0025] Als nächstes sei davon ausgegangen, dass als erstes ein "Empfangsbefehl" ausgeführt wird und danach ein "Sendebefehl" ausgeführt wird.

[0026] Wie es in der Fig. 18 dargestellt ist, führt beispielsweise der Empfangsthread den "Empfangsbefehl" in einem Taktzyklus c7 aus, während der Sendethread den Steuerleitungsport in einem Taktzyklus c9 ausführt.

[0027] Der Empfangsthread sorgt dafür, dass die Spannung am Steuerleitungsport "rrx(O)" im Taktzyklus c7 dem Pegel "HOCH" entspricht. Ferner wartet der Empfangsthread, bis die Spannung am Steuerleitungsport "rtx(I)" auf "HOCH" geht. Wenn die Spannung am Steuerleitungsport "rtx(I)" auf "HOCH" geht, nimmt der Empfangsthread auf die Daten d2 am Datenleitungsport "rdata" Bezug, und er sorgt in einem Taktzyklus c10 dafür, dass die Spannung am Steuerleitungsport "rrx(O)" auf "NIEDRIG" geht.

[0028] Von Chin-tung Chen et al. ist in 'An architectural power optimization case study using high-level synthesis COMPUTER DESIGN: VLSI IN COMPUTERS AND PROCESSORS; 1997; ICCD '97: PROCEEDINGS:; 1997 IEEE INTERNATIONAL CONFERENCE ON AUSTIN, TX, USA, 12-15. OKT. 1997, LOS ALAMITOS, CA, USA, IEEE COMPUT. SOC, US, 12. Oktober 1997 (1997-10-12), Seiten 562-570, XP010251787 ISBN: 0-8186-8206-X eine High-Level-Synthese von Architekturen mit niedrigem Energieverbrauch mit einem Chipdesign offenbart. Zu einem offenbarten Verfahren zur Leistungsoptimierung gehört eine Torschaltung für ein Taktsignal zum Sperren des Taktsignals einer sequenziellen Komponente, wenn diese im Leerlauf arbeitet.

ZUSAMMENFASSUNG DER ERFINDUNG

[0029] Gemäß einer Erscheinungsform der Erfindung eine High-Level-Synthesevorrichtung zum Synthetisieren einer Logikschaltung auf Registertransferlevel aus einer eine Verarbeitungsoperation der Schaltung beschreibenden Verhaltensbeschreibung gemäß dem Anspruch 1.

[0030] Gemäß einer Ausführungsform dieser Erfindung erzeugt der Niederer-Energieverbrauch-Schaltung-Erzeugungsabschnitt die Niederer-Energieverbrauch-Schaltung, die die Taktsignalzufuhr zu den Teilschaltungen stoppt oder sperrt, wenn sich diese in einem Wartezustand befinden, auf Grundlage von Information zu synchronen Verarbeitungsvorgängen, die aus Verhaltensbeschreibungsinformation entnommen wird, die Information zu synchroner Kommunikation enthält.

[0031] Gemäß einer anderen Erscheinungsform der Erfindung ist ein High-Level-Syntheseverfahren zum Synthetisieren einer Logikschaltung auf Registertransferlevel aus einer Verhaltensbeschreibung, die eine Verarbeitungsoperation durch die Logikschaltung beschreibt, um eine gewünschte Logikschaltung aufzubauen, und zum Erzeugen der synthetisierten Logikschaltung gemäß dem Anspruch 3 geschaffen. Bei einer Ausführungsform dieser Erfindung stoppt oder sperrt die synthetisierte Niederer-Energieverbrauch-Schaltung die Operationen von Teilschaltungen durch Stoppen oder Herabsetzen der Taktsignallieferung an die Teilschaltungen.

[0032] Bei einer Ausführungsform dieser Erfindung steuert die synthetisierte Niederer-Energieverbrauch-Schaltung die Taktsignalversorgung unter Verwendung eines Signals, das den Wartezustand anzeigt.

[0033] Bei einer Ausführungsform dieser Erfindung

erzeugt die synthetisierte Niederer-Energieverbrauch-Schaltung ein den Wartezustand der Teilschaltungen anzeigendes Signal unter Verwendung des Sendeanforderungssignals und des Empfangsanforderungssignals, und sie betreibt die Teilschaltungen unter Verwendung eines geschalteten Taktsignals, wie es unter Verwendung des den Wartezustand der Teilschaltungen anzeigenden Signals erzeugt wird, um einen niedrigen Energieverbrauch zu erzielen.

[0034] Bei einer Ausführungsform wird der Synthetisierschritt auf Grundlage von Information zu synchroner Verarbeitung ausgeführt, die aus Information zu synchroner Kommunikation enthaltender Verhaltensbeschreibungsinformation entnommen wird, und die synthetisierte Niederer-Energieverbrauch-Schaltung stoppt oder sperrt die Operationen von Teilschaltungen durch Stoppen oder Sperren der Taktsignalversorgung an die Teilschaltungen.

[0035] Bei dieser Ausführungsform dieser Erfindung verfügt das High-Level-Syntheseverfahren ferner über die Schritte des Erzeugens eines Signals für jede Teilschaltung, das anzeigt, dass sich diese in einem Wartezustand befindet, auf Grundlage von Information zu synchroner Verarbeitung, die aus die Information zu synchroner Kommunikation enthaltender Verhaltensbeschreibungsinformation entnommen wird, und des Erzeugens eines geschalteten Taktsignals für jede Teilschaltung unter Verwendung dieses Signals. Die synthetisierte Niederer-Energieverbrauch-Schaltung betreibt die Teilschaltung unter Verwendung des erzeugten geschalteten Taktsignals, und sie stoppt die Ausgabe des an die Teilschaltung gelieferten geschalteten Taktsignals, wenn sich die Teilschaltung in einem Wartezustand befindet.

[0036] Gemäß einer anderen Erscheinungsform der Erfindung ist ein Verfahren zum Herstellen einer Logikschaltung geschaffen, bei dem das oben beschriebene High-Level-Syntheseverfahren zum Konzipieren der Logikschaltung verwendet wird.

[0037] Gemäß einer anderen Erscheinungsform der Erfindung verfügt ein Computerlesbarer Aufzeichnungsträger über ein Steuerungsprogramm zum Ausführen des oben beschriebenen High-Level-Syntheseverfahrens.

[0038] So ermöglicht es die hier beschriebene Erfindung, die folgenden Vorteile zu erzielen: Bereitstellen einer High-Level-Synthesevorrichtung, die eine Niederer-Energieverbrauch-Schaltungskonfiguration schaffen kann, bei der der vergeudete Energieverbrauch, wenn sich ein Thread in einem Wartezustand befindet, verringert werden kann, eines High-Level-Syntheseverfahrens unter Verwendung der High-Level-Synthesevorrichtung, eines Verfahrens zum Herstellen einer Logikschaltung unter Verwendung des High-Level-Syntheseverfahrens, und eines Computer-lesbaren Aufzeichnungsträgers, auf dem ein Steuerungsprogramm zum Ausführen des High-Level-Syntheseverfahrens aufgezeichnet ist.

[0039] Diese und andere Vorteile der Erfindung werden dem Fachmann beim Lesen und Verstehen der folgenden detaillierten Beschreibung unter Bezugnahme auf die beigefügten Figuren ersichtlich werden.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

[0040] Fig. 1 ist ein Blockdiagramm einer Konfiguration einer High-Level-Synthesevorrichtung gemäß einem Beispiel der Erfindung, wobei Hauptfunktionsabschnitte derselben dargestellt sind.

[0041] Fig. 2A ist ein Diagramm, das eine beispielhafte Verhaltensbeschreibung zeigt.

[0042] Fig. 2B ist ein Diagramm, das Information zu synchroner Verarbeitung in synchronen Kanälen zeigt.

[0043] Fig. 3 ist ein schematisches Diagramm, das ein Ergebnis einer CDFG-Erzeugung durch die High-Level-Synthesevorrichtung der Fig. 1 zeigt.

[0044] Fig. 4 ist ein schematisches Diagramm, das das Ergebnis einer Zeitplanung durch die High-Level-Synthesevorrichtung der Fig. 1 zeigt.

[0045] Fig. 5 ist ein schematisches Diagramm, das ein Ergebnis einer Zuordnung (Berechnungseinheit) durch die High-Level-Synthesevorrichtung der Fig. 1 zeigt.

[0046] Fig. 6 ist ein schematisches Diagramm, das ein Ergebnis einer Zuordnung (Register) durch die High-Level-Synthesevorrichtung der Fig. 1 zeigt.

[0047] Fig. 7 ist ein schematisches Diagramm, das ein Ergebnis bei der Erzeugung von Datenpfaden durch die High-Level-Synthesevorrichtung der Fig. 1 zeigt.

[0048] Fig. 8 ist ein schematisches Diagramm, das ein Ergebnis bei der Erzeugung eines Steuerungsmechanismus durch die High-Level-Synthesevorrichtung der Fig. 1 zeigt.

[0049] Fig. 9A ist ein schematisches Diagramm, das ein Ergebnis beim Erzeugen einer einem Thread 1 entsprechenden Niederer-Energieverbrauch-Schaltung durch die High-Level-Synthesevorrichtung der Fig. 1 zeigt.

[0050] Fig. 9B ist ein schematisches Diagramm, das ein Ergebnis beim Erzeugen einer einem Thread

2 entsprechenden Niederer-Energieverbrauch-Schaltung durch die High-Level-Synthesevorrichtung der Fig. 1 zeigt.

[0051] Fig. 10 ist ein schematisches Diagramm, das ein Ergebnis beim Erzeugen einer Schaltung auf RT-Level durch die High-Level-Synthesevorrichtung der Fig. 1 zeigt.

[0052] Fig. 11 ist ein Diagramm, das ein beispielhaftes Verfahren mit geschaltetem Taktsignal zeigt.

[0053] Fig. 12 ist ein Schaltbild für den Fall, dass die Niederer-Energieverbrauch-Schaltung gemäß der Erfindung bei beispielhafter synchroner Kommunikation, wie sie durch die Fig. 17 veranschaulicht ist, angewandt wird.

[0054] Fig. 13 ist ein Timingdiagramm, das ein Beispiel mit der Niederer-Energieverbrauch-Schaltung der Fig. 12 zeigt.

[0055] Fig. 14 ist ein Timingdiagramm, das eine beispielhafte Operation der Schaltung auf RT-Level gemäß der Fig. 10 zeigt.

[0056] Fig. 15 ist ein Diagramm, das eine beispielhafte Verhaltensbeschreibung einschließlich synchroner Kommunikation zeigt.

[0057] Fig. 16 ist ein Blockdiagramm einer Konfiguration einer herkömmlichen High-Level-Synthesevorrichtung, wobei Hauptfunktionsabschnitte derselben dargestellt sind.

[0058] Fig. 17 ist ein schematisches Schaltbild, das ein beispielhaftes Verfahren zum Realisieren synchroner Kommunikation zeigt.

[0059] Fig. 18 ist ein Timingdiagramm, das eine beispielhafte Operation der synchronen Kommunikation gemäß der Fig. 17 zeigt.

BESCHREIBUNG DER BEVORZUGTEN AUSFÜHRUNGSFORMEN

[0060] Nachfolgend werden eine High-Level-Synthesevorrichtung und ein High-Level-Syntheseverfahren gemäß der Erfindung unter Bezugnahme auf die beigefügten Zeichnungen beschrieben.

[0061] Die Fig. 1 ist ein Blockdiagramm einer Konfiguration einer High-Level-Synthesevorrichtung gemäß einem Beispiel der Erfindung, wobei Hauptfunktionsabschnitte derselben dargestellt sind. In der Fig. 1 verfügt eine High-Level-Synthesevorrichtung **100** über einen Verhaltensbeschreibungs-Speicherabschnitt **20**, einen Verhaltensbeschreibungs-Analysierabschnitt **3**, einen Threadunterteilungsabschnitt **4**, einen Speicherabschnitt **5** für Information für synchrone Verarbeitung, einen CDFG(control data flow graph)-Erzeugungsabschnitt **6**, einen Zeitplanungsabschnitt **7**, einen Zuordnungsabschnitt **8**, einen Datenpfaderzeugungsabschnitt **9**, einen Steuerungsmechanismus-Erzeugungsabschnitt **10**, einen Niederer-Energieverbrauch-Schaltung-Erzeugungsabschnitt **11**, einen Beschreibungserzeugungsabschnitt **12** für eine Schaltung auf RT-Level sowie einen Speicherabschnitt **13** für eine Schaltung auf RT-Level.

[0062] Der Verhaltensbeschreibungs-Speicherabschnitt **20** speichert eine Verhaltensbeschreibung (beispielsweise die Fig. 2A) einer Logikschaltung (digitaler LSI-Schaltkreis), durch die ein Algorithmus für ein Verarbeitungsverfahren entsprechend einer High-Level-Synthese angegeben ist.

[0063] Der Verhaltensbeschreibungs-Analysierabschnitt **3** entnimmt und analysiert eine Verhaltensbeschreibung, wie in der Fig. 2A dargestellt, die einen Algorithmus für die Verarbeitungsoperation einer Schaltung beschreibt, um in der Verhaltensbeschreibung enthaltene Information zu synchroner Verarbeitung zu entnehmen. Die Information zu synchroner Verarbeitung ist Information zu einem Prozess, der Synchronisation zwischen Threads erfordert. Beispielsweise betrifft derartige Information einen Kommunikationsvorgang unter Verwendung eines synchronen Kanals. Genauer gesagt, enthält die Information zu synchroner Verarbeitung nicht nur Information zu synchroner Verarbeitung für einen synchronen Kanal, sondern auch Information zu anderen Prozessen. Zur Information zu anderen Prozessen gehört beispielsweise Information zu einem Synchronisationsprozess, gemäß dem mehrere Blöcke in einem Parallelsatz, der parallel arbeitet, die Verarbeitung gleichzeitig starten, oder wobei ein nächster Prozess im Parallelsatz gestartet wird, nachdem alle Blöcke beendet sind.

[0064] Der Threadunterteilungsabschnitt **4** folgt einem Syntaxbaum, und er unterteilt diesen in mehrere Threads, die asynchron und parallel abgearbeitet werden, und er erzeugt eine Liste von Einzelinformationen zu synchronen Kanälen, über die die Threads eine Operation "Senden" ausführen, und eine Liste von Einzelinformationen zu synchronen Kanälen, über die die Threads eine Operation "Empfangen" ausführen. Es sei darauf hingewiesen, dass bei der Verhaltensbeschreibung in der Fig. 2A die Zeilen 7-14 einem Thread 1 entsprechen und die Zeilen 15-19 einem Thread 2 entsprechen.

[0065] Der Speicherabschnitt **5** für Information zu synchroner Verarbeitung speichert Information zu einem synchronen Prozess betreffend einen synchronen Kanal, wie es in der Fig. 2B dargestellt ist, wie für jeden Thread durch den Threadunterteilungsabschnitt **4** erzeugt, oder andere Information zu synchroner Verarbeitung.

[0066] Der CDFG-Erzeugungsabschnitt **6** erzeugt ein CDFG (control data flow graph), wie es beispielsweise in der Fig. 3 dargestellt ist, auf Grundlage der Verhaltensbeschreibung in der Fig. 2A. Zugehörige Einzelheiten werden unten beschrieben.

[0067] Der Zeitplanungsabschnitt **7** erzeugt einen Zeitplan, gemäß dem ein CDFG in mehrere Schritte unterteilt wird, wie es beispielhaft in der Fig. 4 dargestellt ist.

[0068] Der Zuordnungsabschnitt **8** ordnet Berechnungseinheiten, wie es in der Fig. 5 dargestellt ist, und Register, wie es in der Fig. 6 dargestellt ist, zu einem CDFG zu, wie es unten beschrieben ist.

[0069] Der Datenpfaderzeugungsabschnitt **9** erzeugt (verbindet) Datenbusse (Datenpfade) zwischen Schaltungselementen, die den Berechnungseinheiten oder Registern zugeordnet sind, wie es durch dicke Linien in der Fig. 7 dargestellt ist.

[0070] Der Steuerungsmechanismus-Erzeugungsabschnitt **10** fügt zu einer Schaltung mit Datenbusverbindungen einen Steuerungsmechanismus hinzu, wie es in der Fig. 8 mit dicken Linien dargestellt ist. Der Steuerungsmechanismus ist mit Steuerungsleitungen verbunden.

[0071] Der Niederer-Energieverbrauch-Schaltung-Erzeugungsabschnitt **11** erzeugt beispielsweise Niederer-Energieverbrauch-Schaltungen **11A** und **11B**, wie sie in den Fig. 9A und Fig. 9B dargestellt sind, auf Grundlage der Information zu synchroner Verarbeitung betreffend die synchronen Kanäle, wie sie im in der Fig. 2B dargestellten Speicherabschnitt **5** für Information zu synchroner Verarbeitung abgespeichert ist. Die Niederer-Energieverbrauch-Schaltungen **11A** und **11B** stoppen die Zufuhr eines Taktsignals zu einer Teilschaltung für jeden Thread, wenn sich dieser in einem Wartezustand befindet.

[0072] Der Beschreibungserzeugungsabschnitt **12** für einer Schaltung auf RT-Level (Erzeugungsabschnitt für eine Schaltung auf RT-Level) verbindet Teilschaltungen (Threads 1 und 2) und die Niederer-Energieverbrauch-Schaltungen **11A** und **11B**, beispielsweise gemäß der Fig. 10, um eine Schaltung auf RT-Level zu erzeugen.

[0073] Der Speicherabschnitt **13** für eine Schaltung auf RT-Level speichert erzeugte Information zu einer Schaltung auf RT-Level.

[0074] Nun wird als beispielhaftes Verfahren zum Steuern der an eine Schaltung zu liefernden Taktsignale ein Verfahren mit geschaltetem Taktsignal unter Verwendung von UND-Logikelementen oder ODER-Logikelementen beschrieben. Bei diesem Verfahren werden, wie es in der Fig. 11 dargestellt ist, durch Ändern eines Steuerungssignals (Aktivierungssignals) Taktsignalimpulse mit einem gewünschten Timing ausgegeben. Das Verfahren mit geschaltetem Taktsignal kann als beispielhaftes Verfahren zum Stoppen des Betriebs einer Teilschaltung verwendet werden. Es können andere Verfahren zum Stoppen des Betriebs einer Teilschaltung verwendet werden.

[0075] Wenn beispielsweise das Verbindung mit geschaltetem Taktsignal bei der in der Fig. 17 dargestellten Schaltung angewandt wird, ist diese so konfiguriert, wie es in der Fig. 12 dargestellt ist. Ein Timingdiagramm dieser Schaltung ist dergestalt, wie es in der Fig. 13 dargestellt ist. Wie es im Timingdiagramm der Fig. 13 dargestellt ist, wird das Taktsignal für den Thread 1 (Senden) in den Taktzyklen c2 und c3 gestoppt, während ein Taktsignal für den Thread 2 (Empfangen) in den Taktzyklen c8 und c9 gestoppt wird. Daher kann der durch Taktsignale verursachte Energieverbrauch auf 9/11 gesenkt werden.

[0076] Demgemäß wird, bei der Erfindung, die erzeugte Niederer-Energieverbrauch-Schaltung dazu verwendet, die Taktsignallieferung an eine Teilschaltung zu stoppen, wenn sich ein Thread in einem Wartezustand befindet (auf hohem oder niedrigem Pegel gehalten wird). Im Ergebnis ist die Schaltrate der Teilschaltung verringert, wodurch ein niedriger Energieverbrauch erzielt wird.

[0077] Nun wird eine Betriebsweise der oben beschriebenen Konfiguration detailliert beschrieben.

[0078] Als erstes sei angenommen, dass eine Verhaltensbeschreibung vorliegt, bei der ein Algorithmus für einen Prozess zur High-Level-Synthese so angegeben ist, wie es in der Fig. 2A dargestellt ist. In diesem Fall analysiert der Verhaltensbeschreibungs-Analyseabschnitt **3** eine vom Verhaltensbeschreibungs-Speicherabschnitt **20** erhaltene Verhaltensbeschreibung. Der Threadunterteilungsabschnitt **4** ordnet dem Thread 1 einen ersten Block im Parallelsatz der Verhaltensbeschreibung in der Fig. 2A zu, und er ordnet dem Thread 2 einen zweiten Block zu. Synchrone Kanäle, auf die die Threads 1 und 2 zugreifen, sind dergestalt, wie es beispielhaft in der Fig. 2B (Tabelle 1) dargestellt ist. Information zu den synchronen Kanälen ist im Speicherabschnitt **5** für Information zu synchroner Verarbeitung abgespeichert.

[0079] Als nächstes erzeugt der CDFG-Erzeugungsabschnitt **6** ein CDFG (control data flow graph). Beispielsweise wird auf Grundlage der Verhaltensbeschreibung der Fig. 2A ein CDFG (control data flow graph), wie es in der Fig. 3 dargestellt ist, erzeugt, bei dem Eingaben aus einem Kommunikationskanal oder Ausgaben an diesen, wie "Senden" oder "Empfangen (rcv)" oder dergleichen, verschiedene Rechenvorgänge, wie Addition "+", Subtraktion "–", Mul-

tiplikation "x" und dergleichen durch Knoten repräsentiert sind und Datenabhängigkeitsbeziehungen zwischen jeweiligen Knoten durch direkte Zweige repräsentiert sind.

[0080] Ferner führt der Zeitplanungsabschnitt **7** einen Zeitplanungsprozess aus. Beispielsweise werden die Knoten im Thread 1 in der Fig. 3 Schritten 1 bis 4 zugeordnet, und die Knoten im Thread 2 werden den Schritten 1 und 2 zugeordnet, wodurch ein Zeitplan erhalten wird, wie er in der Fig. 4 dargestellt ist.

[0081] Ferner ordnet der Zuordnungsabschnitt **8** Schaltungselemente, wie Berechnungseinheiten und Register, den Knoten zu. Zur Zuordnung der Berechnungseinheiten wird, wie es in der Fig. 5 dargestellt ist, "Senden" oder "Empfangen (rcv)" den "Kommunikationssteuerleitungsports" oder den Datenleitungsports "rtx", "rrx", "wtx" und "wrx" zugeordnet, während Addition "+", Subtraktion "–" und Multiplikation "x" einem Addierer "ADD", einem Subtrahierer "SUB" und einem Multiplizierer "MULT" zugeordnet werden.

[0082] Zur Zuordnung der Register werden Zweige über die jeweiligen Schritte hinweg Registern "regx" bis "regz" und "ragA" zugeordnet, was zu Zuordnungen führt, wie sie in der Fig. 6 dargestellt sind.

[0083] Ferner erzeugt der Datenpfaderzeugungsabschnitt **9** Datenbusse (Datenpfade). Auf Grundlage der in den Fig. 5 und Fig. 6 dargestellten Prozesse werden Datenbusse erzeugt, wie sie in der Fig. 7 mit dicken Linien dargestellt sind.

[0084] Ferner erzeugt der Steuerungsmechanismus-Erzeugungsabschnitt **10** Steuerungsleitungen zum Steuern eines Steuerungsmechanismus "CTRL1", eines Steuerungsmechanismus "CTRL2", Kommunikation sowie Registern "regx" bis "regz" und "rage".

[0085] Fernre erzeugt der Niederer-Energieverbrauch-Schaltung-Erzeugungsabschnitt **11**, der ein charakteristischer Teil der Erfindung ist, die Niederer-Energieverbrauch-Schaltung **11A** und **11B**, wie sie in den Fig. 9A und Fig. 9B dargestellt sind. Hierbei ist angenommen, dass an die Threads 1 und 2 gelieferte Taktsignale durch ein Verfahren mit geschaltetem Taktsignal gesteuert werden, bei dem ein einer ODER-Logik entsprechendes Gatter als Torschaltungselement für ein Taktsignal verwendet wird. Auf Grundlage von Kanalinformation, wie sie durch den Threadunterteilungsabschnitt entnommen wird, auf den jeder Thread 1 und 2 zugreift, werden die folgenden Prozesse (1) bis (3) ausgeführt:

(1) für einen Lesekanal "rch" wird unter Verwendung einer Kommunikationssteuerungsleitung "rrx_rch" und einer Steuerungsleitung n"rtx_rch" die folgende Logik

wait_rch ≤ rrx_rch AND(not rtx_rch)

erzeugt;
(2) für einen Schreibkanal "wch" wird unter Verwendung einer Kommunikationssteuerungsleitung "wtx_rch" und einer Steuerungsleitung "wrx_rch" die folgende Logik

wait_wch ≤ wtx_rch AND(not wrx_wch)

erzeugt;
(3) aus den erzeugten Logikwerten und dem Taktsignal "clk" wird das geschaltete Taktsignal "gclk" erzeugt. Hierbei wird, wenn ein Thread mehrere Lesekanäle und/oder Schreibkanäle enthält, die logische ODER-Verknüpfung von "wait_rch" und/oder "wait_wch", entsprechend jedem Kanal, berechnet.

glck ≤ clk OR (wait_rch oder wait_wch);

gemäß der Fig. 2B (Tabelle 1) werden die in der **Fig.** 9 dargestellten Schaltungen **11A** und **11B** zum Erzeugen geschalteter Taktsignale erzeugt.

[0086] Abschließend verbindet der Beschreibungserzeugungsabschnitt **12** für eine Schaltung auf RT-Level Teilschaltungen in den Threads 1 und 2 sowie die Niederer-Energieverbrauch-Schaltungen **11A** und **11B** miteinander, was zu einer Schaltung auf RT-Level führt, wie sie in der Fig. 10 dargestellt ist.

[0087] Die Fig. 14 zeigt ein beispielhaftes Timingdiagramm für die Schaltung auf RT-Level gemäß der Fig. 10.

[0088] In diesem Fall ist eine Operation des Threads 1 die folgende.

[0089] Wie es in der Fig. 14 dargestellt ist, versucht der Thread 1 als erstes, Daten von einem synchronen Kanal I in einem Taktzyklus c1 zu empfangen (rrx_i ist auf "HOCH" gesetzt), da jedoch keine Daten nach außen gesendet werden, wartet der Thread 1 bis zu einem Taktzyklus c6.

[0090] Als nächstes empfängt der Thread 1 im Taktzyklus c6 Daten x1 vom synchronen Kanal i.

[0091] In einem Taktzyklus c7 empfängt der Thread 1 Daten y1 vom synchronen Kanal i, um eine Berechnung auszuführen.

[0092] In einem Taktzyklus c8 wird das Ergebnis x1 der Berechnung an einen synchronen Kanal ch geliefert. Diese Daten werden vom Thread 2 empfangen, und unmittelbar danach wird die Kommunikation beendet.

[0093] Im Taktzyklus c9 wird erneut eine Emp-

fangsanforderung an den synchronen Kanal i gesendet.

[0094] In einem Taktzyklus c11 empfängt der Thread 1 Daten x2 vom synchronen Kanal i.

[0095] In einem Taktzyklus c12 empfängt der Thread 1 Daten y2 vom synchronen Kanal i, um eine Berechnung auszuführen.

[0096] In einem Taktzyklus c13 wird das Ergebnis z2 der Berechnung an den synchronen Kanal ch geliefert und durch den Thread 2 empfangen.

[0097] In einem Taktzyklus c14 wird eine Empfangsanforderung zur Außenseite der Schaltung geliefert.

[0098] Eine Operation des Threads 2 ist die folgende.

[0099] Als erstes gibt der Thread 2, im Taktzyklus c1, eine Empfangsanforderung an einen synchronen Kanal ch aus, und er wartet, bis der Thread 1 im Taktzyklus c9 Daten sendet.

[0100] Als nächstes empfängt der Thread 2 im Taktzyklus c9 Daten z1 vom synchronen Kanal ch, und er liefert die Daten an einen synchronen Kanal 0.

[0101] Im Taktzyklus c12 wird der Sendevorgang im synchronen Kanal 0 beendet, und der Thread 2 gibt erneut eine Empfangsanforderung an den synchronen Kanal ch aus.

[0102] Im Taktzyklus C14 empfängt der Thread Daten z1 vom synchronen Kanal ch, und er sendet sie an den synchronen Kanal 0.

[0103] Wie es in der Fig. 14 dargestellt ist, treten bei einem Taktsignalverfahren, bei dem dauernd ein Taktsignal geliefert wird, in jedem der Threads 1 und 2 14 Taktsignaloperationen auf, so dass die Lastkapazität innerhalb eines durch ein Taktsignal betriebenen Schaltkreises geladen oder entladen wird und Energie verbraucht wird.

[0104] Jedoch sei angenommen, dass dann, wenn sich die Threads 1 und 2 bei synchroner Kommunikation in einem Wartezustand befinden, die Taktsignalversorgung gesteuert wird. Gemäß dem Beispiel sind, wenn das Verfahren mit geschaltetem Taktsignal verwendet wird, die Anzahlen von Taktsignalen (die Anzahlen der Anstiege der Taktsignale), wie sie an den Thread 1 und den Thread 2 geliefert werden, acht bzw. drei.

[0105] Daher sind die durch die Taktsignale für die Threads 1 und 2 verursachten Energieverbräuche entsprechend den Faktoren 8/14 bzw. 3/14 im Vergleich zu einem Verfahren verringert, bei dem dauernd ein Taktsignal geliefert wird. So wird durch das High-Level-Syntheseverfahren gemäß der Erfindung die durch Taktsignale verursachte Schaltrate um den Faktor N/T verringert, wobei T die Anzahl der Taktzyklen beim Betrieb einer Teilschaltung ist und N die Anzahl der Taktzyklen in einem Wartezustand ist.

[0106] Ein herkömmliches Schaltungsdesign auf RT-Level und Gatterpegel benötigt Änderungen der Designdaten, wie einer HDL-Beschreibung, einer Netzliste oder dergleichen, um das Verfahren mit geschaltetem Taktsignal zu verwenden. Demgegenüber ermöglicht es das Verfahren mit niedrigem Energieverbrauch gemäß der Erfindung, ursprüngliche Designdaten (eine Verhaltensbeschreibung) ohne jede Änderung zu verwenden, wodurch eine Periode zum Entwerfen eines niedrigen Energieverbrauchs beseitigt wird.

[0107] Es sei darauf hingewiesen, dass es zwar beim oben beschriebenen Beispiel nicht speziell angegeben ist, dass jedoch anschließend eine Logikschaltungskonfiguration automatisch als Designdaten durch das High-Level-Syntheseverfahren unter Verwendung der High-Level-Synthesevorrichtung gemäß der Erfindung aus einer Verhaltensbeschreibung konstruiert wird. Der Energieverbrauch der erzeugten Logikschaltung selbst kann durch eine Niederer-Energieverbrauch-Schaltung gesenkt werden. Dies ist der Effekt der vorliegenden Erfindung.

[0108] Ferner ist es zwar beim oben beschriebenen Beispiel nicht speziell angegeben, jedoch verfügt die Hardwarekonfiguration der High-Level-Synthesevorrichtung **100** mit dem Verhaltensbeschreibungs-Analysierabschnitt **3**, dem Threadunterteilungsabschnitt **4**, dem CDFG-Erzeugungsabschnitt **6**, dem Zeitplanungsabschnitt **7**, dem Zuordnungsabschnitt **8**, dem Datenpfaderzeugungsabschnitt **9**, dem Steuerungsmechanismus-Erzeugungsabschnitt **10**, dem Niederer-Energieverbrauch-Schaltung-Erzeugungsabschnitt **11**, dem Beschreibungserzeugungsabschnitt **12** für eine Schaltung auf RT-Level und dem Speicherabschnitt **13** für eine Schaltung auf RT-Level ferner über folgendes: einen Eingabeabschnitt, der von einem Benutzer dazu verwendet wird, Bedienungsinhalte einzugeben (eine Tastatur, eine Maus und dergleichen), einen Speicherabschnitt, der ein Steuerungsprogramm zum Steuern des High-Level-Syntheseverfahrens gemäß der Erfindung und die zugehörigen Daten speichert (einen RAM als Arbeitsspeicher und einen ROM als lesbares Aufzeichnungsmedium), eine Datenbank (den Verhaltensbeschreibungs-Speicherabschnitt **20** und den Speicherabschnitt **5** für Information zu synchroner Verarbeitung oder dergleichen), und einen Steuerungsabschnitt, der durch eingegebene Operationsbefehle vom Eingabeabschnitt gesteuert wird und das High-Level-Syntheseverfahren auf Grundlage des Steue-

rungsprogramms im ROM ausführt (CPU; zentrale Verarbeitungseinheit). Ferner verfügt die Hardwarekonfiguration der High-Level-Synthesevorrichtung **100** über einen Anzeigeabschnitt, wie ein Flüssigkristalldisplay und dergleichen, zum Anzeigen eines Anfangsschirms, eines Anleitungsschirms, eines Ausführungsergebnisses und dergleichen.

[0109] Ferner wird, beim oben angegebenen Beispiel, eine Niederer-Energieverbrauch-Schaltung so synthetisiert und konfiguriert, dass dann, wenn sich eine Teilschaltung (ein Thread) in einem Wartezustand befindet, der Betrieb der Teilschaltung gestoppt wird, wodurch im Ergebnis die Schaltrate α abgesenkt wird, um den Energieverbrauch zu verringern. Die Erfindung ist nicht hierauf eingeschränkt. Der Energieverbrauch P einer CMOS-Logikschaltung ist proportional zu ihrer Schaltrate α, ihrer Lastkapazität C, ihrer Betriebsspannung V^2 und ihrer Betriebsfrequenz f. Daher kann alternativ, wenn sich eine Teilschaltung (Thread) in einem Wartezustand befindet, eine Niederer-Energieverbrauch-Schaltung synthetisiert werden, die die Betriebsspannung der Teilschaltung sperrt. In diesem Fall kann der Energieverbrauch gesenkt werden. Insbesondere kann eine Niederer-Energieverbrauch-Schaltung so synthetisiert und konfiguriert werden, dass dann, wenn sich eine Teilschaltung in einem Wartezustand befindet, der Spannungspegel eines an die Teilschaltung gelieferten Taktsignals gesperrt wird. In diesem Fall kann eine Niederer-Energieverbrauch-Schaltung synthetisiert werden, bei der der durch ein Taktsignal verursachte Energieverbrauch verringert werden kann.

[0110] Wie oben beschrieben, wird, gemäß der Erfindung, wenn sich eine Teilschaltung in einem Wartezustand befindet, der Betrieb der Teilschaltung gestoppt oder gesperrt. Daher kann die Schaltrate oder der Betriebsspannungspegel der Teilschaltung verringert werden. Ferner kann, da der Energieverbrauch einer Logikschaltung proportional zu ihrer Schaltrate oder ihrem Betriebsspannungspegel ist, eine Niederer-Energieverbrauch-Schaltung mit verringertem Energieverbrauch synthetisiert werden.

[0111] Ferner wird dann, wenn sich eine Teilschaltung in einem Wartezustand befindet, die Taktsignalzufuhr zur Teilschaltung gestoppt oder gesperrt. Daher kann eine Niederer-Energieverbrauch-Schaltung synthetisiert werden, bei der der durch ein Taktsignal verursachte Energieverbrauch, wie er anfällt, wenn sich die Teilschaltung in einem Wartezustand befindet, gesenkt werden kann.

[0112] Ferner wird dann, wenn sich eine Teilschaltung während eines Datenübertragungsvorgangs in einem Wartezustand befindet, der Betrieb der Teilschaltung gestoppt oder gesperrt. Daher kann eine Niederer-Energieverbrauch-Schaltung synthetisiert werden, bei der ihre Schaltrate oder ihr Betriebsspannungspegel verringert ist, um den Energieverbrauch zu senken.

[0113] Ferner kann eine Niederer-Energieverbrauch-Schaltung synthetisiert werden, bei der dann, wenn sich eine Teilschaltung in einem Wartezustand befindet, ein geschaltetes Taktsignal, gemäß dem kein Impuls erzeugt wird, dazu verwendet wird, die Teilschaltung zu betreiben, so dass der durch das Taktsignal verursachte Energieverbrauch gesenkt werden kann.

[0114] Ferner kann die Taktsignalzufuhr gestoppt oder gesperrt werden, wenn sich eine Teilschaltung in einem Wartezustand befindet. Daher kann eine Niederer-Energieverbrauch-Schaltung konzipiert werden, bei der der Energieverbrauch, wie er auf Grund eines Taktsignals im Wartezustand anfällt, gesenkt werden kann.

[0115] Ferner kann die Taktsignalzufuhr für jede Teilschaltung gestoppt oder gesperrt werden, wenn sich die Teilschaltung in einem Wartezustand befindet. Daher kann eine Niederer-Energieverbrauch-Schaltung synthetisiert werden, bei der der durch ein Taktsignal verursachte Energieverbrauch, wie er anfällt, wenn sich jede Teilschaltung in einem Wartezustand befindet.

[0116] Ferner kann eine Logikschaltung mit einer Niederer-Energieverbrauch-Schaltung erhalten werden, wenn ein High-Level-Syntheseverfahren zum Erzeugen einer Niederer-Energieverbrauch-Schaltung, bei der die Energie verringert ist, wie sie nutzlos verbraucht wird, wenn sich eine Teilschaltung in einem Wartezustand befindet, zum Konzipieren einer Logikschaltung verwendet wird und die konzipierte Logikschaltung hergestellt wird.

[0117] Ferner kann ein auf einem Aufzeichnungsträger gespeichertes Steuerungsprogramm dazu verwendet werden, das High-Level-Syntheseverfahren für eine Niederer-Energieverbrauch-Schaltung, bei der nutzlos verbrauchte Energie gesenkt werden kann, zu realisieren.

Patentansprüche

1. High-Level-Synthesevorrichtung zum Synthetisieren einer Registertransferlevel-Logikschaltung mit einer ersten Teilschaltung (Thread 1) und einer zweiten Teilschaltung (Thread 2) aus einer Verhaltensbeschreibung, die eine Verarbeitungsoperation der Schaltung beschreibt, wobei diese Vorrichtung mit Folgendem versehen ist:
einem Niederer-Energieverbrauch-Schaltung-Erzeugungsabschnitt zum Erzeugen einer Niederer-Energieverbrauch-Schaltung, die die Schaltungsoperationen der die Logikschaltung bildenden ersten

und/oder zweiten Teilschaltung nur dann stoppt oder sperrt, wenn sich die erste und/oder zweite Teilschaltung in einem Wartezustand befindet, um niedrigen Energieverbrauch zu erzielen;
wobei der Niederer-Energieverbrauch-Schaltung-Erzeugungsabschnitt gemeinsam mit der Logikschaltung synthetisiert wird;
wobei die erste Teilschaltung eine Datensenderendeschaltung aufweist und die zweite Teilschaltung eine Datenempfängerendeschaltung aufweist;
die Datensenderendeschaltung und die Datenempfängerendeschaltung jeweils über einen Sendeanforderungs-Kontrollleitungsport (wtx, rtx), einen Empfangsanforderungs-Kontrollleitungsport (wrx, rrx) und einen Datenleitungsport (wdata, rdata) verfügen; wobei der Sendeanforderungs-Kontrollleitungsport und der Datenleitungsport der Datenempfängerendeschaltung mit dem Sendeanforderungs-Kontrollleitungsport bzw. dem Datenleitungsport der Datensenderendeschaltung verbunden sind, um ein Sendeanforderungssignal bzw. Daten zu empfangen, wie sie am Sendeanforderungs-Kontrollleitungsport bzw. am Datenleitungsport der Datensenderendeschaltung bereitgestellt werden; und der Empfangsanforderungs-Kontrollleitungsport der Datensenderendeschaltung mit dem Empfangsanforderungs-Kontrollleitungsport der Datenempfängerendeschaltung verbunden ist, um ein Empfangsanforderungssignal zu empfangen, wie es am Empfangsanforderungs-Kontrollleitungsport der Datenempfängerendeschaltung bereitgestellt wird;
wobei die Datensenderendeschaltung so konfiguriert ist, dass sie das Sendeanforderungssignal erzeugt, das anzeigt, dass sie an ihrem Datenleitungsport Daten bereitgestellt hat, und um dieses Sendeanforderungssignal als hohe Spannung an ihrem Sendeanforderungs-Kontrollleitungsport bereit zu stellen;
und wobei die Datenempfängerendeschaltung so konfiguriert ist, dass sie das Empfangsanforderungssignal erzeugt, das anzeigt, dass sie dazu bereit ist, die Daten von der Datensenderendeschaltung über ihren Datenleitungsport zu empfangen, und um das Empfangsanforderungssignal als hohe Spannung an ihrem Empfangsanforderungs-Kontrollleitungsport bereit zu stellen;
dadurch gekennzeichnet, dass die synthetisierte Niederer-Energieverbrauch-Schaltung so konfiguriert ist, dass sie ein Eingangssignal von den Sendeanforderungs-Kontrollleitungsports und den Empfangsanforderungs-Kontrollleitungsports der Datensenderendeschaltung und der Datenempfängerendeschaltung empfängt;
dass die synthetisierte Niederer-Energieverbrauch-Schaltung so konfiguriert ist, dass sie bestimmt, dass sich die Datensenderendeschaltung in einem Wartezustand befindet, wenn eine Spannung an ihrem Sendeanforderungs-Kontrollleitungsport hoch ist und eine Spannung an ihrem Empfangsanforderungs-Kontrollleitungsport niedrig ist;
dass die synthetisierte Niederer-Energieverbrauch-Schaltung so konfiguriert ist, dass sie bestimmt, dass sich die Datenempfängerendeschaltung in einem Wartezustand befindet, wenn eine Spannung an ihrem Empfangsanforderungs-Kontrollleitungsport hoch ist und eine Spannung an ihrem Sendeanforderungs-Kontrollleitungsport niedrig ist;
und dass die synthetisierte Niederer-Energieverbrauch-Schaltung die Operationen der ersten Teilschaltung stoppt oder sperrt, wenn sich die Datensenderendeschaltung in einem Wartezustand befindet, und sie die Operationen der zweiten Teilschaltung stoppt oder sperrt, wenn sich die Datenempfängerendeschaltung in einem Wartezustand befindet, während eine Übertragung der Daten zwischen den Teilschaltungen erfolgt.

2. High-Level-Synthesevorrichtung nach Anspruch 1, bei der der Niederer-Energieverbrauch-Schaltung-Erzeugungsabschnitt die Niederer-Energieverbrauch-Schaltung erzeugt, die die Taktsignallieferung an die Teilschaltungen stoppt oder sperrt, wenn sich diese im genannten Wartezustand befinden, was auf Grundlage von Synchronverarbeitungsinformation erfolgt, die aus Verhaltensbeschreibungsinformation entnommen wird, die Synchronkommunikationsinformation enthält.

3. High-Level-Syntheseverfahren zum Synthetisieren einer Registertransferlevel-Logikschaltung mit einer ersten Teilschaltung (Thread 1) und einer zweiten Teilschaltung (Thread 2) aus einer Verhaltensbeschreibung, die eine Verarbeitungsoperation der Schaltung beschreibt, wobei dieses Verfahren den folgenden Schritt aufweist:
Synthetisieren gemeinsam mit der Logikschaltung, einer Niederer-Energieverbrauch-Schaltung, die die Schaltungsoperationen der die Logikschaltung bildenden ersten und/oder zweiten Teilschaltung nur dann stoppt oder sperrt, wenn sich die erste und/oder zweite Teilschaltung in einem Wartezustand befindet, um niedrigen Energieverbrauch zu erzielen;
wobei die erste Teilschaltung eine Datensenderendeschaltung aufweist und die zweite Teilschaltung eine Datenempfängerendeschaltung aufweist;
die Datensenderendeschaltung und die Datenempfängerendeschaltung jeweils über einen Sendeanforderungs-Kontrollleitungsport (wtx, rtx), einen Empfangsanforderungs-Kontrollleitungsport (wrx, rrx) und einen Datenleitungsport (wdata, rdata) verfügen; wobei der Sendeanforderungs-Kontrollleitungsport und der Datenleitungsport der Datenempfängerendeschaltung mit dem Sendeanforderungs-Kontrollleitungsport bzw. dem Datenleitungsport der Datensenderendeschaltung verbunden sind, um ein Sendeanforderungssignal bzw. Daten zu empfangen, wie sie am Sendeanforderungs-Kontrollleitungsport bzw. am Datenleitungsport der Datensenderendeschaltung bereitgestellt werden; und der Empfangsanforderungs-Kontrollleitungsport der Datensenderendeschaltung mit dem Empfangsanforderungs-Kontroll-

leitungsport der Datenempfängerendeschaltung verbunden ist, um ein Empfangsanforderungssignal zu empfangen, wie es am Empfangsanforderungs-Kontrollleitungsport der Datenempfängerendeschaltung bereitgestellt wird;
wobei die Datensenderendeschaltung so konfiguriert ist, dass sie das Sendeanforderungssignal erzeugt, das anzeigt, dass sie an ihrem Datenleitungsport Daten bereitgestellt hat, und um dieses Sendeanforderungssignal als hohe Spannung an ihrem Sendeanforderungs-Kontrollleitungsport bereit zu stellen;
und wobei die Datenempfängerendeschaltung so konfiguriert ist, dass sie das Empfangsanforderungssignal erzeugt, das anzeigt, dass sie dazu bereit ist, die Daten von der Datensenderendeschaltung über ihren Datenleitungsport zu empfangen, und um das Empfangsanforderungssignal als hohe Spannung an ihrem Empfangsanforderungs-Kontrollleitungsport bereit zu stellen;
dadurch gekennzeichnet, dass die synthetisierte Niederer-Energieverbrauch-Schaltung so konfiguriert ist, dass sie ein Eingangssignal von den Sendeanforderungs-Kontrollleitungsports und den Empfangsanforderungs-Kontrollleitungsports der Datensenderendeschaltung und der Datenempfängerendeschaltung empfängt;
dass die synthetisierte Niederer-Energieverbrauch-Schaltung so konfiguriert ist, dass sie bestimmt, dass sich die Datensenderendeschaltung in einem Wartezustand befindet, wenn eine Spannung an ihrem Sendeanforderungs-Kontrollleitungsport hoch ist und eine Spannung an ihrem Empfangsanforderungs-Kontrollleitungsport niedrig ist;
dass die synthetisierte Niederer-Energieverbrauch-Schaltung so konfiguriert ist, dass sie bestimmt, dass sich die Datenempfängerendeschaltung in einem Wartezustand befindet, wenn eine Spannung an ihrem Empfangsanforderungs-Kontrollleitungsport hoch ist und eine Spannung an ihrem Sendeanforderungs-Kontrollleitungsport niedrig ist;
und dass die synthetisierte Niederer-Energieverbrauch-Schaltung die Operationen der ersten Teilschaltung stoppt oder sperrt, wenn sich die Datensenderendeschaltung in einem Wartezustand befindet, und sie die Operationen der zweiten Teilschaltung stoppt oder sperrt, wenn sich die Datenempfängerendeschaltung in einem Wartezustand befindet, während eine Übertragung der Daten zwischen den Teilschaltungen erfolgt.

4. High-Level-Syntheseverfahren nach Anspruch 3, bei dem die synthetisierte Niederer-Energieverbrauch-Schaltung die Operationen der Teilschaltungen dadurch stoppt oder sperrt, dass sie die Taktsignallieferung (clk) an die Teilschaltungen stoppt oder verringert.

5. High-Level-Syntheseverfahren nach Anspruch 3 oder Anspruch 4, bei dem die synthetisierte Niederer-Energieverbrauch-Schaltung die Taktsignallieferung unter Verwendung eines den Wartezustand anzeigenden Signals (g1, g2) steuert.

6. High-Level-Syntheseverfahren nach Anspruch 3 oder Anspruch 4, bei dem die synthetisierte Niederer-Energieverbrauch-Schaltung unter Verwendung des Sendeanforderungssignals und des Empfangsanforderungssignals ein Signal (g1, g2) erzeugt, das den Wartezustand der Teilschaltungen anzeigt, und sie die Teilschaltungen unter Verwendung eines geschalteten Taktsignals (gclk1, gclk2) ansteuert, die unter Verwendung des den Wartezustand der Teilschaltungen anzeigenden Signals erzeugt wird, um niedrigen Energieverbrauch zu erzielen.

7. High-Level-Syntheseverfahren nach Anspruch 3 oder Anspruch 4, bei dem der Synthetisierschritt auf Grundlage von Synchronverarbeitungsinformation ausgeführt wird, die aus Synchronkommunikationsinformation enthaltender Verhaltensbeschreibungsinformation entnommen wird, und die synthetisierte Niederer-Energieverbrauch-Schaltung die Operationen der Teilschaltungen durch Stoppen oder Sperren der Taktsignallieferung (clk) an die Teilschaltungen stoppt oder sperrt.

8. High-Level-Syntheseverfahren nach Anspruch 7, ferner mit den folgenden Schritten:
Erzeugen eines Signals (g1, g2) auf Grundlage von Synchronverarbeitungsinformation, die aus die Synchronkommunikationsinformation enthaltender Verhaltensbeschreibungsinformation entnommen wird, eines Signals (g1, g2) für jede Teilschaltung, das anzeigt, dass sich diese Teilschaltung in einem Wartezustand befindet; und
Erzeugen eines geschalteten Taktsignals (gclk1, gclk2) für jede Teilschaltung unter Verwendung des Signals;
wobei die synthetisierte Niederer-Energieverbrauch-Schaltung die Teilschaltungen unter Verwendung des erzeugten geschalteten Taktsignals ansteuert und die Ausgabe des an die Teilschaltungen gelieferten geschalteten Taktsignals stoppt, wenn sich die Teilschaltungen in einem Wartezustand befinden.

9. Verfahren zum Herstellen einer Logikschaltung, bei dem ein High-Level-Syntheseverfahren gemäß dem Anspruch 3 zum Entwerfen der Logikschaltung verwendet wird.

10. Computer lesbarer Aufzeichnungsträger mit einem Steuerungsprogramm zum Ausführen eines High-Level-Syntheseverfahrens gemäß dem Anspruch 3.

11. Computerprogramm mit einer Programmcodeeinrichtung zum Ausführen aller Schritte des High-Level-Syntheseverfahrens gemäß einem der Ansprüche 3 bis 8, wenn das Programm auf einem

Computer abgearbeitet wird.

Es folgen 17 Blatt Zeichnungen

Anhängende Zeichnungen

FIG. 1

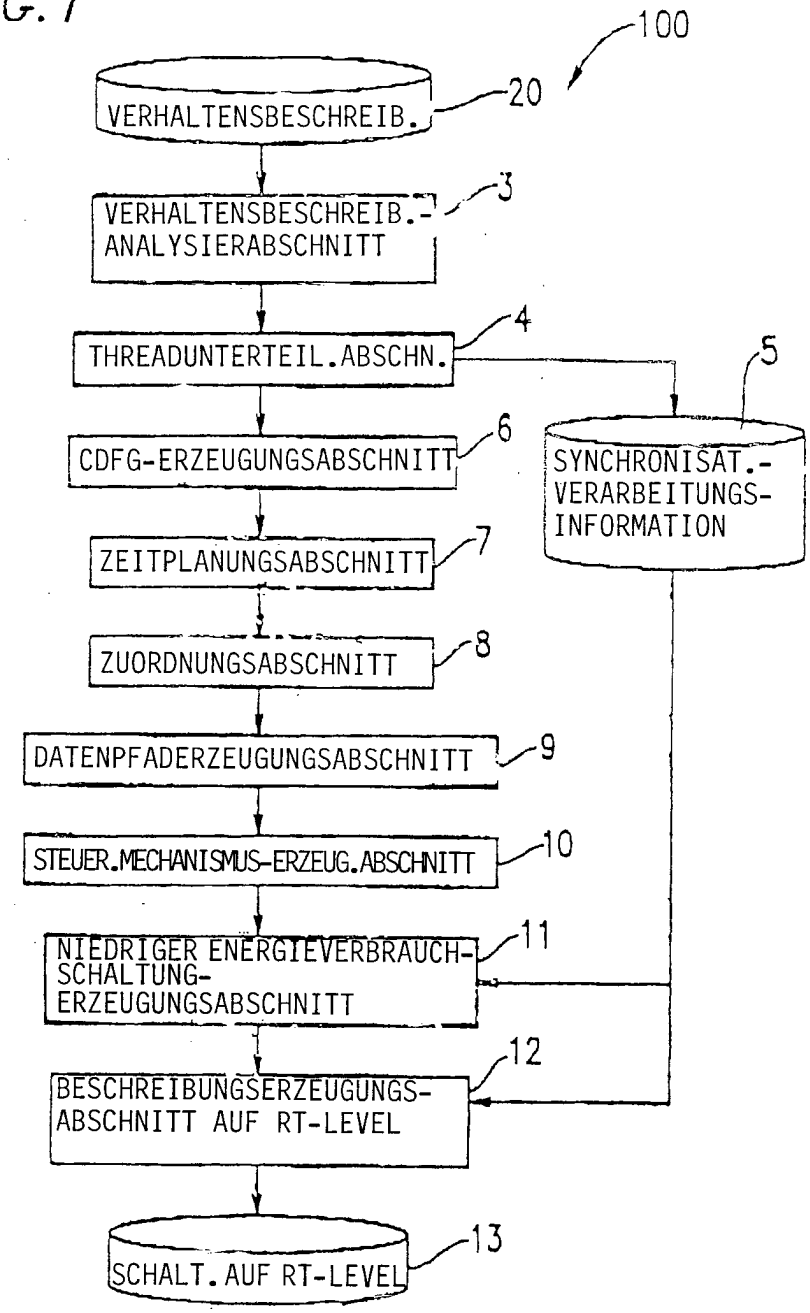


FIG. 2A

```
 1: void f(chan int i, chan int o)
 2: {
 3:   chan int ch;
 4:   int x,y,z;
 5:
 6:   par{
 7:     {/*thread1*/
 8:       while(1){
 9:         x = receive(i);
10:         y = receive(i);
11:         z = (x+y) * (x-y);
12:         send(ch, z);
13:       }
14:     }
15:     {/*thread2*/
16:       while(1){
17:         send(o,receive(ch));
18:       }
19:     }
20:   }
21: }
```

Thread 1

Thread 2

FIG. 2B

	LESEKANAL	SCHREIBKANAL
THREAD 1	i	ch
THREAD 2	ch	o

FIG.3

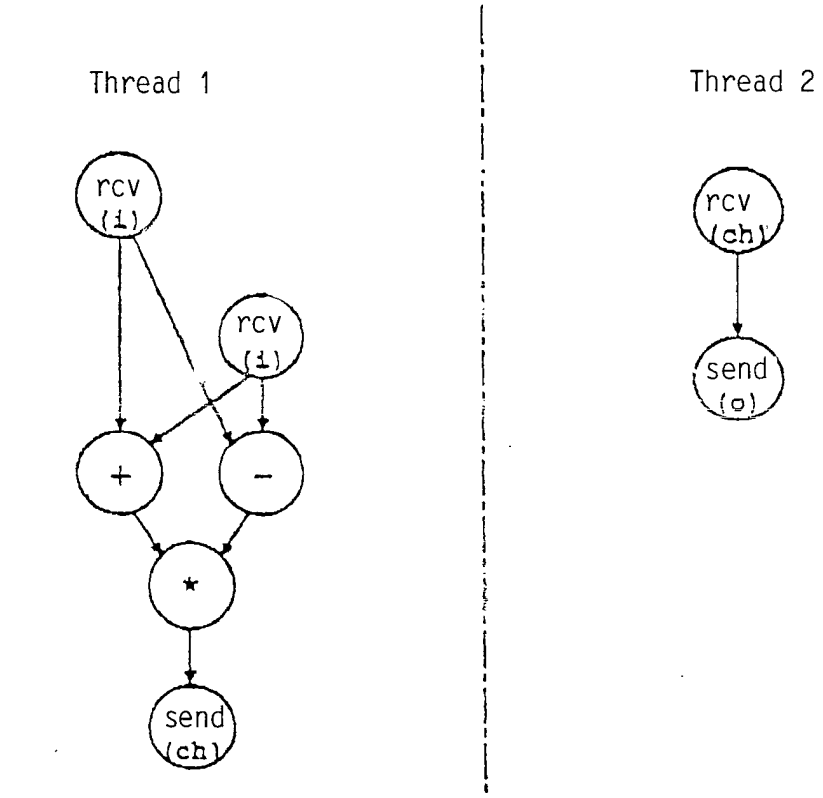


FIG. 4

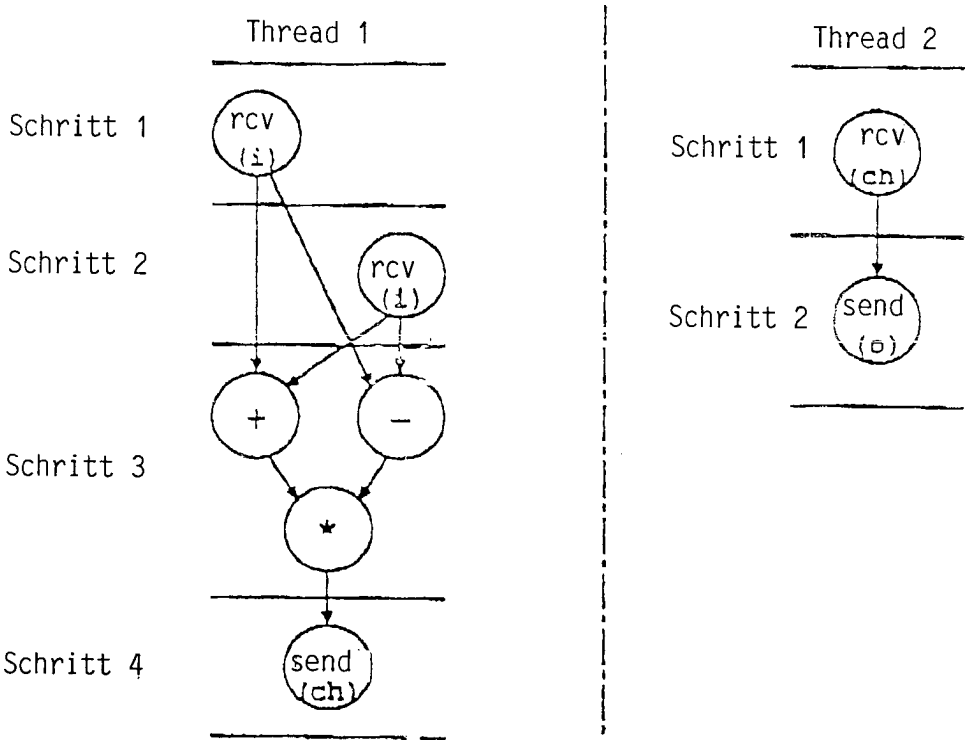

Thread 1
Thread 2
Schritt 1
Schritt 2
Schritt 3
Schritt 4
rcv (i)
rcv (i)
+
−
*
send (ch)
rcv (ch)
send (o)

FIG. 5

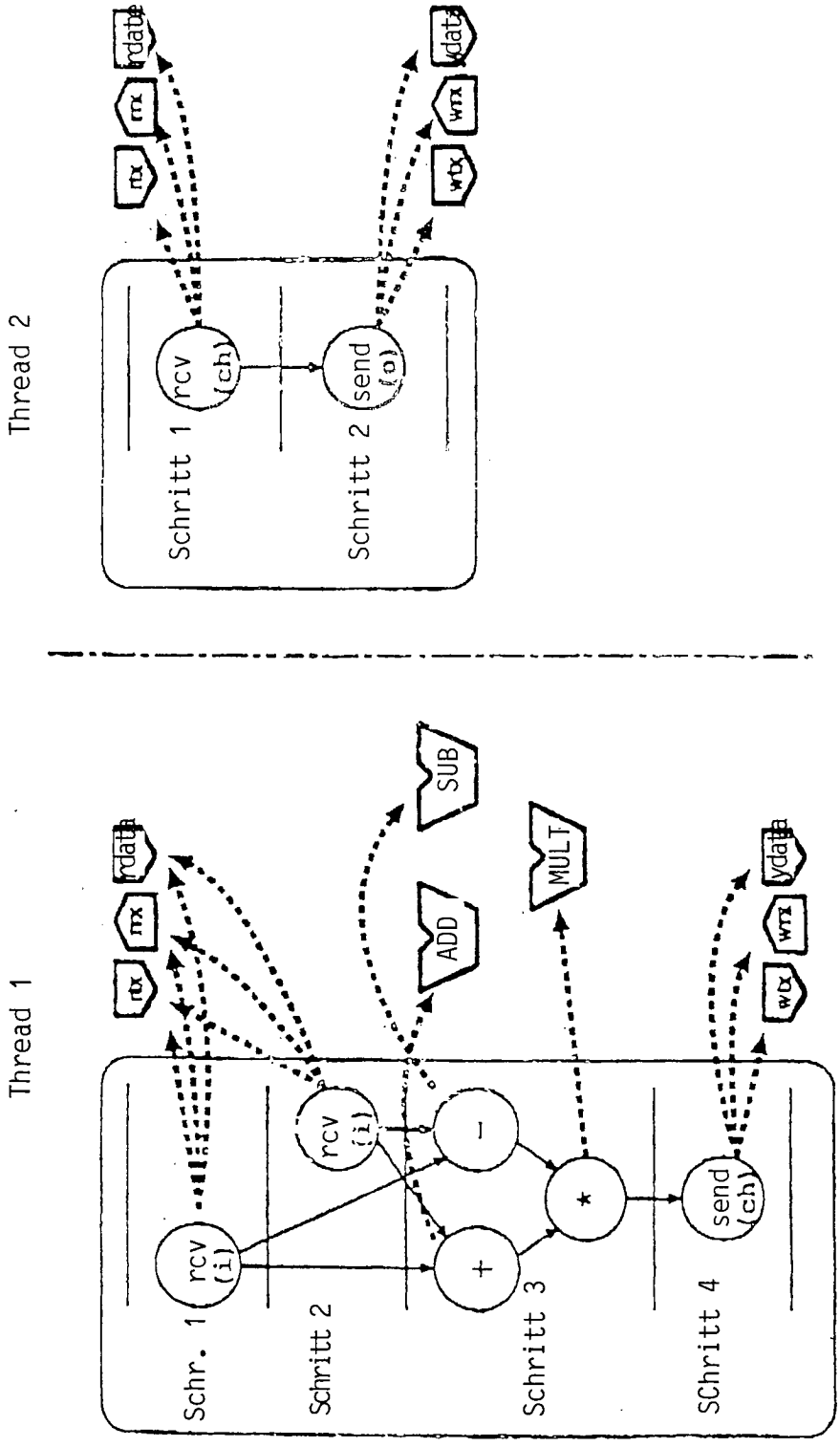

Thread 1
Thread 2
Schr. 1
Schritt 2
Schritt 3
SChritt 4
rcv (i)
rcv (i)
+
-
*
send (ch)
ADD
SUB
MULT
rtx
rrx
rdata
wtx
wrx
ydata
Schritt 1
Schritt 2
rcv (ch)
send (o)

FIG. 6

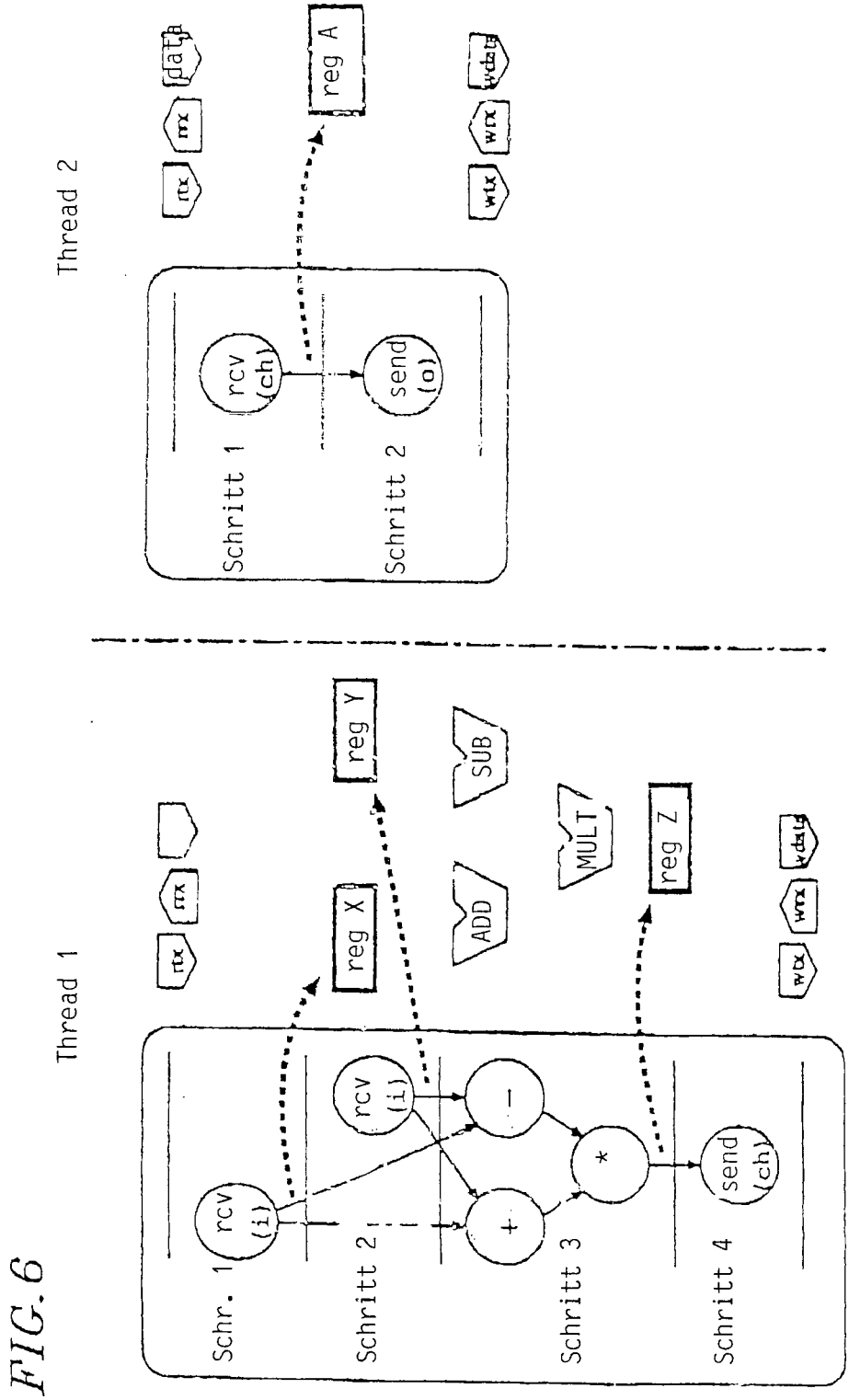

Thread 1
Thread 2
Schr. 1
Schritt 2
Schritt 3
Schritt 4
rcv (i)
rcv (i)
rcv (ch)
send (o)
send (ch)
reg X
reg Y
reg Z
reg A
ADD
SUB
MULT
+
−
*
rtx
wtx
wrx
wdata
data

FIG. 7

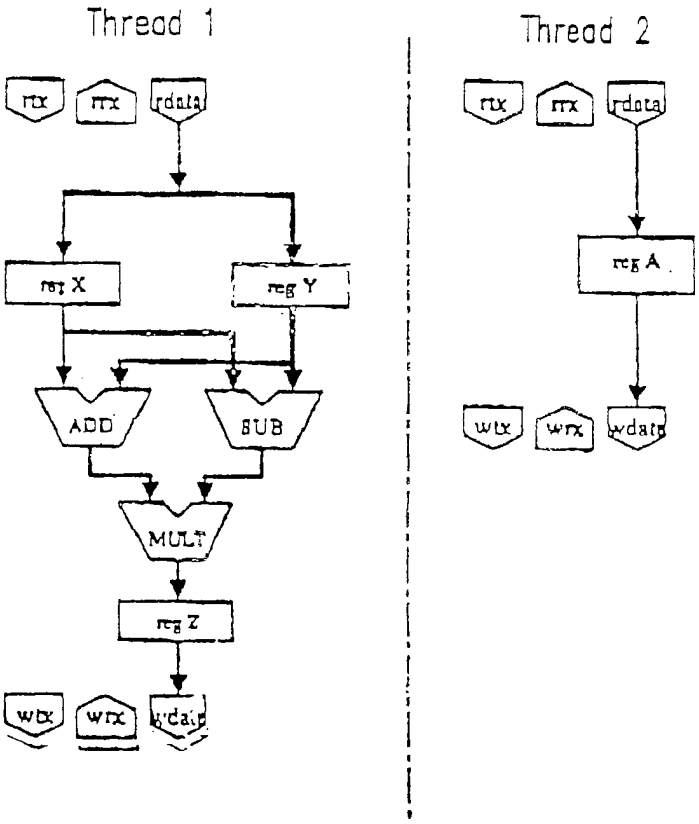

Thread 1
Thread 2
rtx
rtx
rdata
reg X
reg Y
ADD
SUB
MULT
reg Z
wtx
wrx
wdata
rtx
rtx
rdata
reg A
wtx
wrx
wdata

FIG. 8

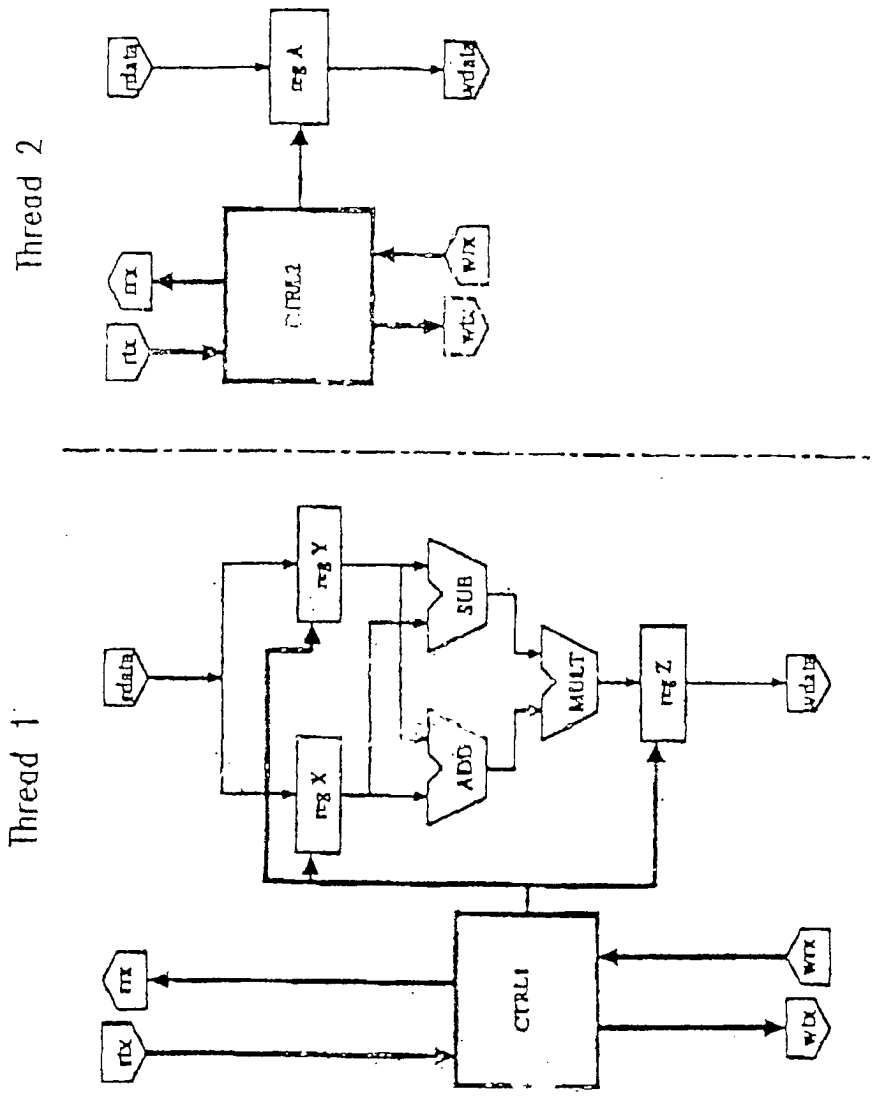

Thread 2
rdata
reg A
wdata
rtx
CTRL2
wtx
Thread 1
reg Y
SUB
rdata
MULT
reg Z
wdata
ADD
reg X
rtx
CTRL1
wtx

FIG. 9A

Thread 1

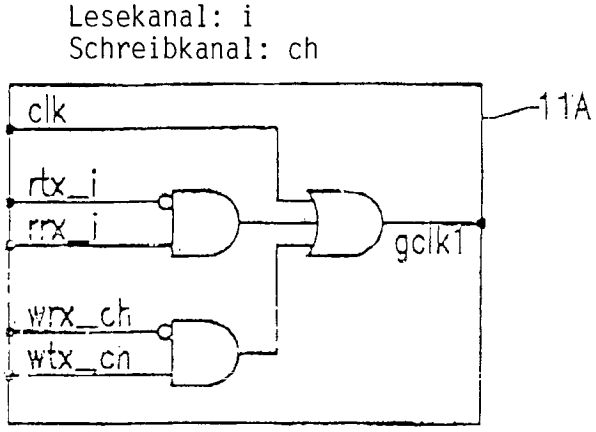


FIG. 9B

Thread 2

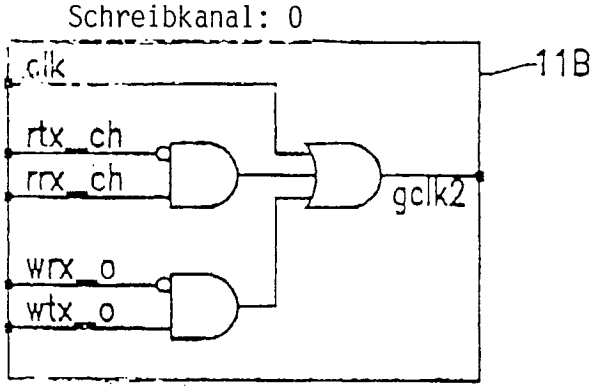


FIG. 10

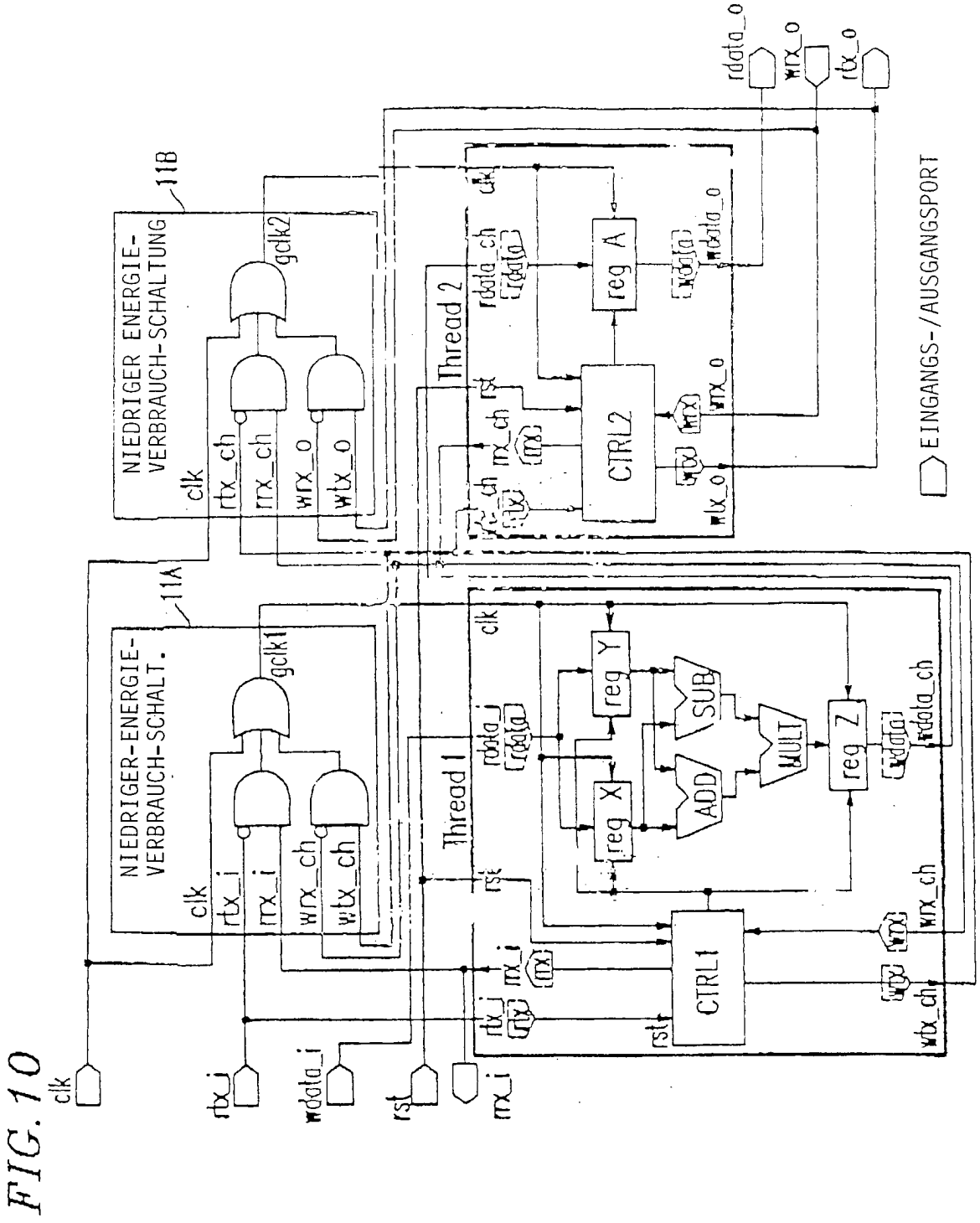

clk
rtx_i
wdata_i
rst
rx_i
NIEDRIGER ENERGIE-VERBRAUCH-SCHALTUNG
11B
11A
gclk2
gclk1
rtx_ch
rrx_ch
wrx_o
wtx_o
wrx_ch
wtx_ch
rtx_i
rrx_i
Thread 1
Thread 2
rdata_i
rdata_ch
rdata_o
reg X
reg Y
reg Z
reg A
ADD
SUB
MULT
CTRL1
CTRL2
wrx_ch
wtx_ch
rtx_o
wrx_o
EINGANGS-/AUSGANGSPORT

FIG. 11

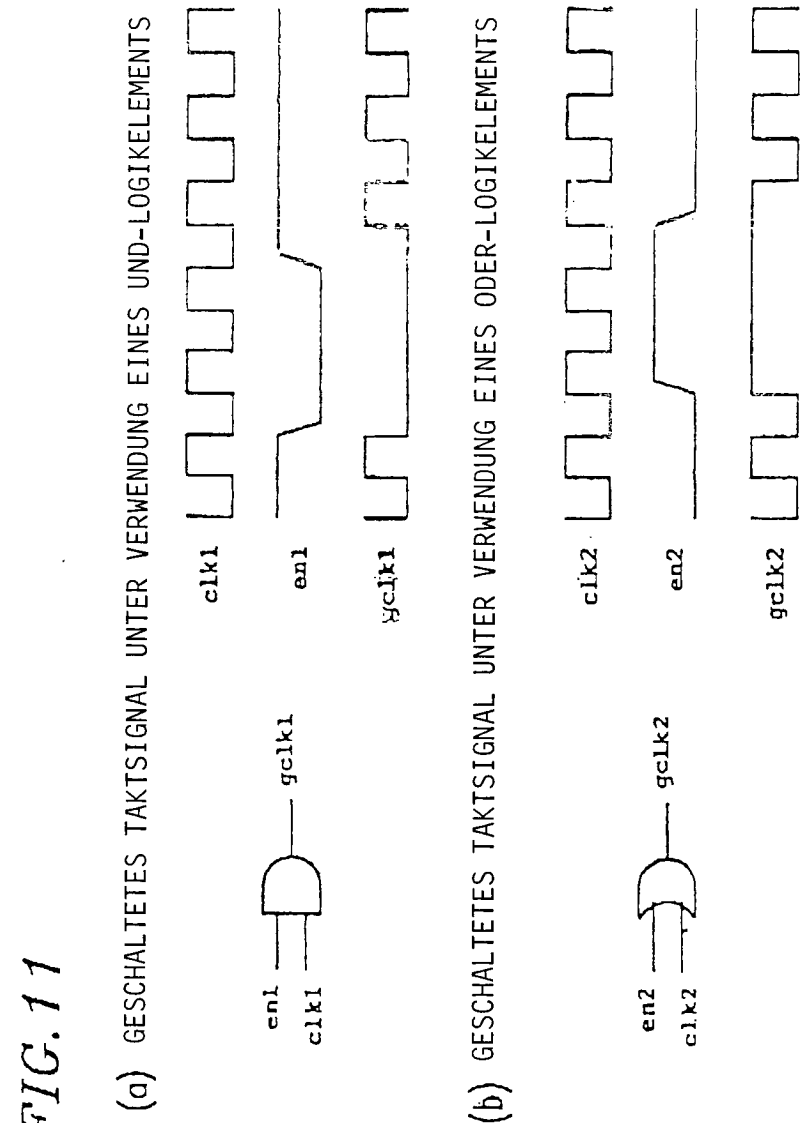

(a) GESCHALTETES TAKTSIGNAL UNTER VERWENDUNG EINES UND-LOGIKELEMENTS
clk1
en1
gclk1
en1
clk1
(b) GESCHALTETES TAKTSIGNAL UNTER VERWENDUNG EINES ODER-LOGIKELEMENTS
clk2
en2
gclk2
en2
clk2

FIG. 12

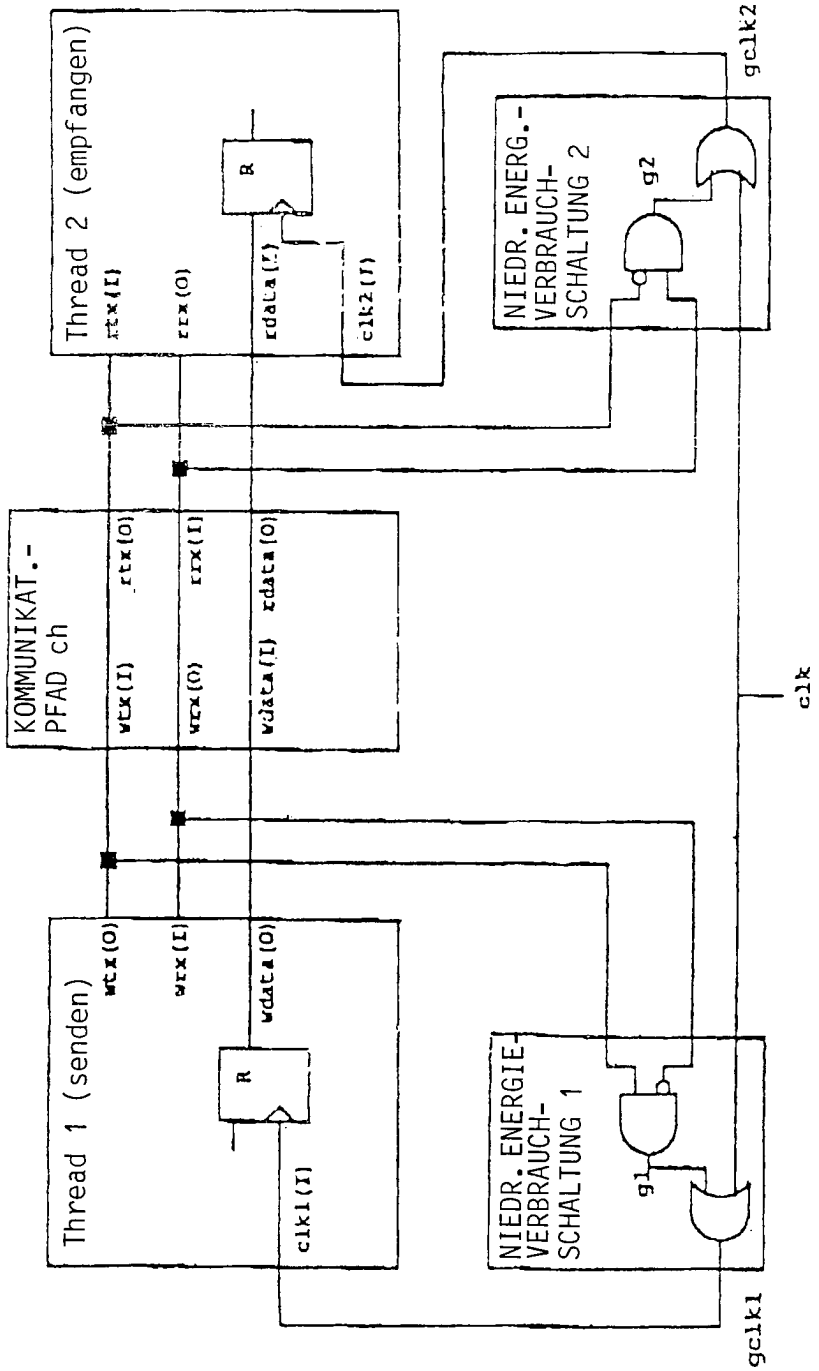

Thread 2 (empfangen)
rtx(I)
rtx(O)
rdata(I)
clk2(I)
R
NIEDR. ENERG.-VERBRAUCH-SCHALTUNG 2
g2
gclk2
KOMMUNIKAT.-PFAD ch
wtx(I)
rtx(O)
wtx(O)
rtx(I)
wdata(I)
rdata(O)
clk
Thread 1 (senden)
wtx(O)
wtx(I)
wdata(O)
R
clk1(I)
NIEDR. ENERGIE-VERBRAUCH-SCHALTUNG 1
g1
gclk1

FIG. 13

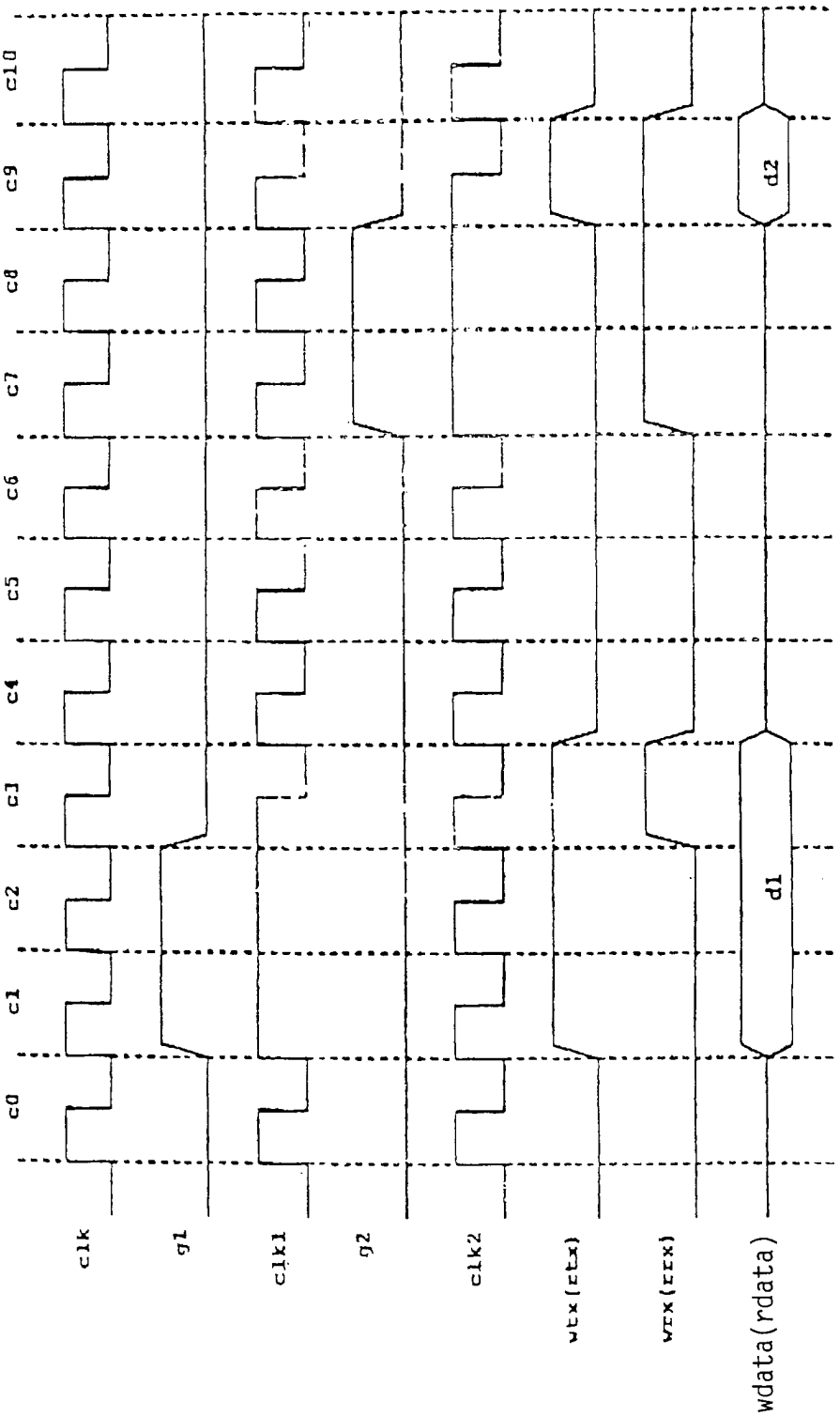

c0
c1
c2
c3
c4
c5
c6
c7
c8
c9
c10
clk
g1
clk1
g2
clk2
wtx(ctx)
wrx(crx)
wdata(rdata)
d1
d2

FIG. 14

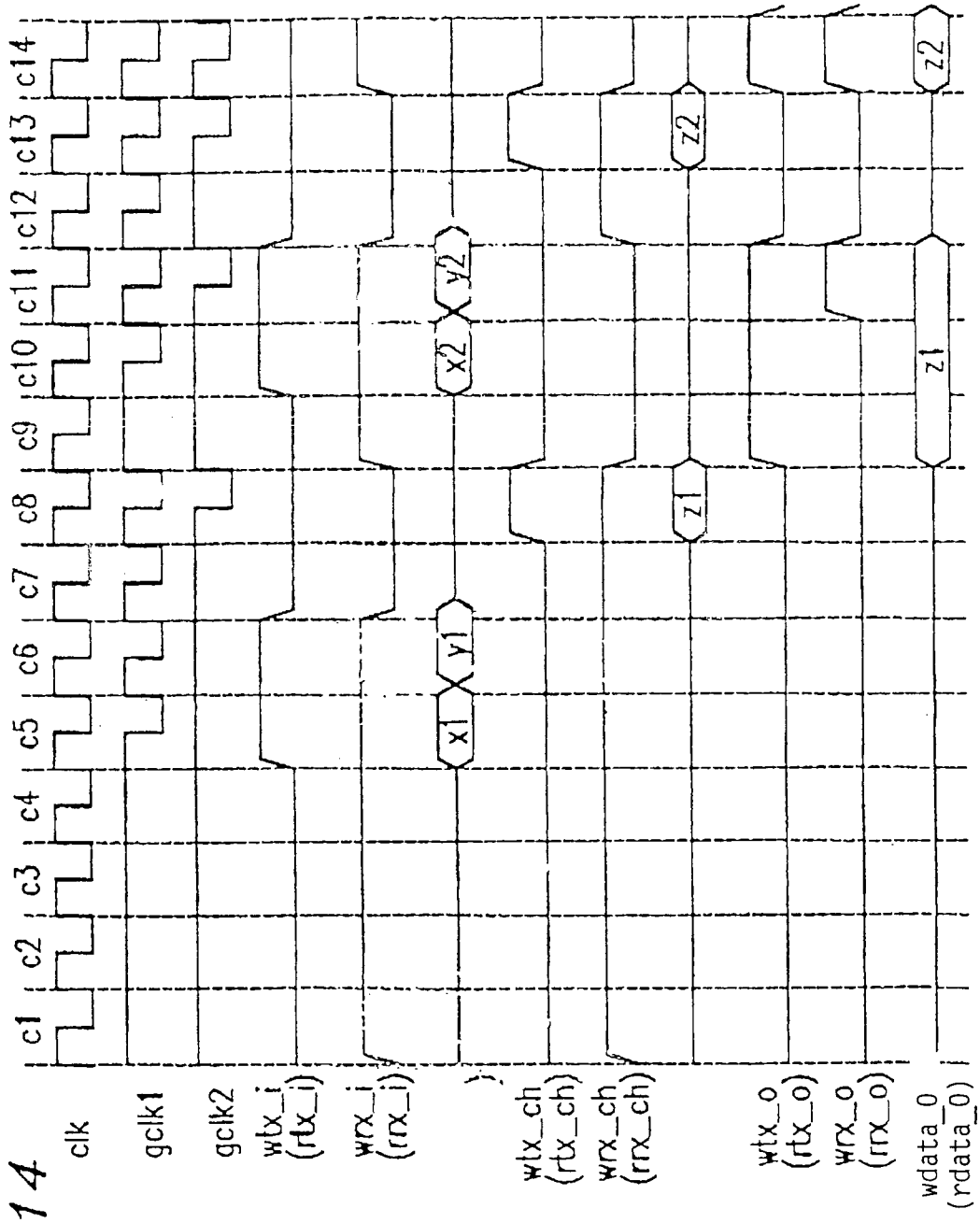
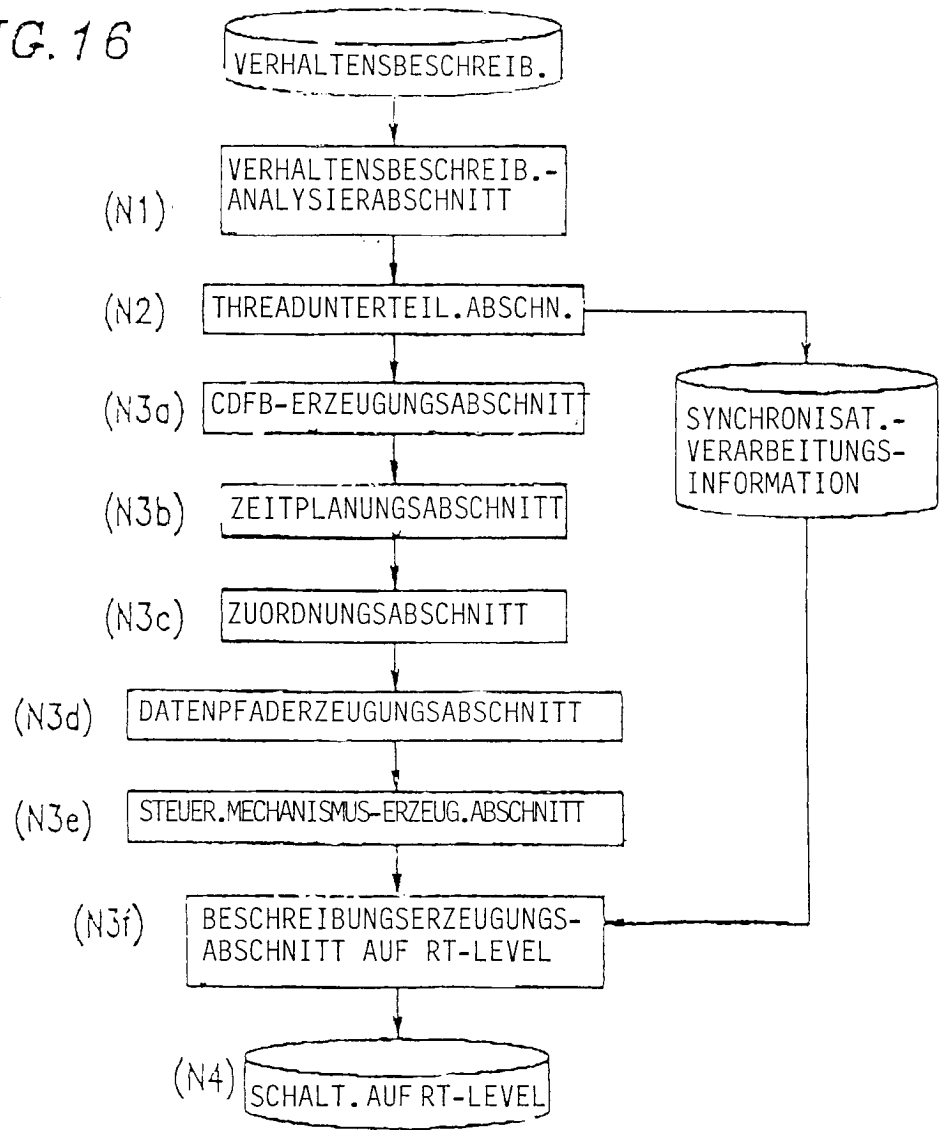

c1
c2
c3
c4
c5
c6
c7
c8
c9
c10
c11
c12
c13
c14
clk
gclk1
gclk2
wtx_i (rtx_i)
wrx_i (rrx_i)
wtx_ch (rtx_ch)
wrx_ch (rrx_ch)
wtx_o (rtx_o)
wrx_o (rrx_o)
wdata_0 (rdata_0)
x1
y1
x2
y2
z1
z2
z1
z2

FIG. 15

```
1: func()
2: {
3:   chan int ch;
4:   int x;
5:
6:   par{
7:     { send(ch, 10);}
8:     { x=receive(ch);}
9:   }
10:  putint(stdout,10,0,x);
11: }
```

FIG. 16



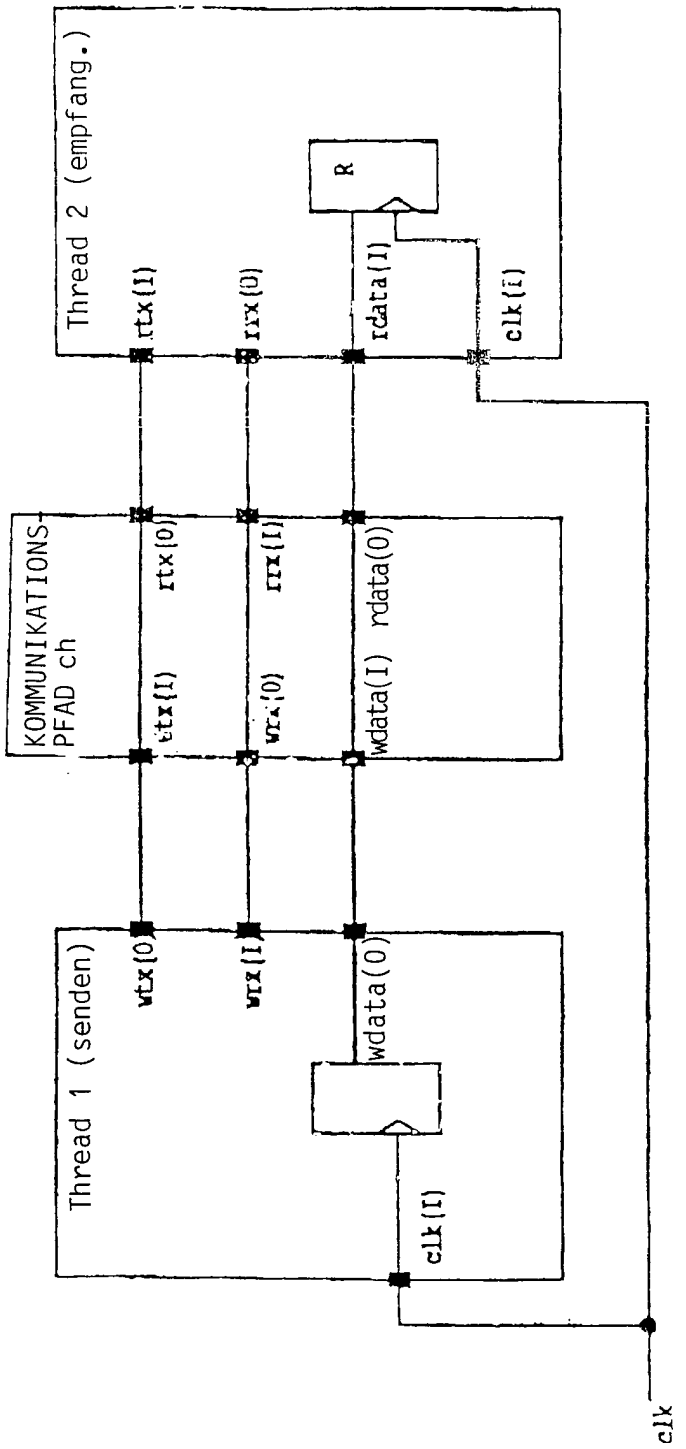
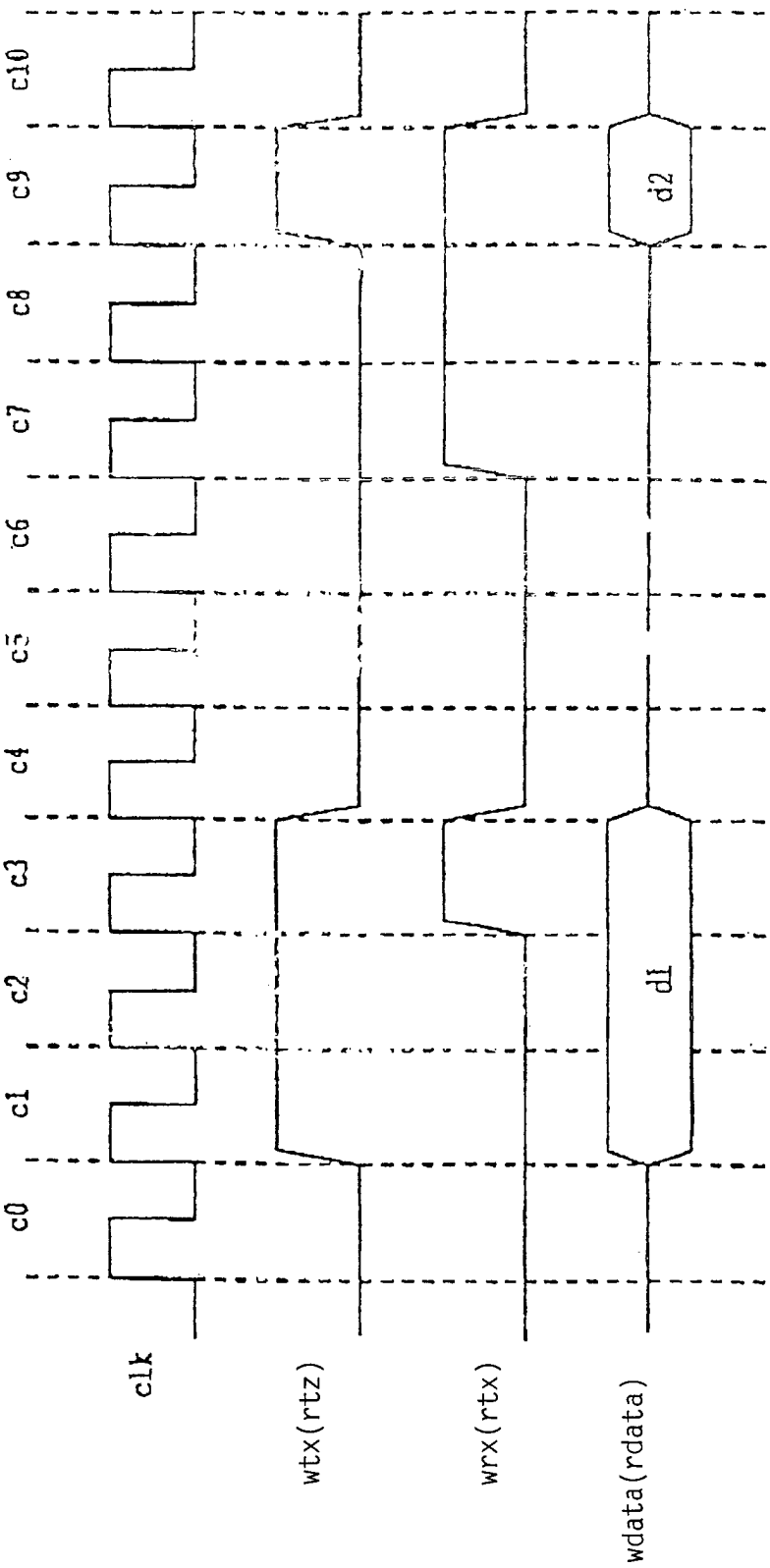

Thread 2 (empfang.)
R
rtx(1)
rtx(0)
rdata(1)
clk(1)
KOMMUNIKATIONS-
PFAD ch
rtx(0)
rtx(1)
rdata(0)
wdata(1)
Thread 1 (senden)
wtx(0)
wtx(1)
wdata(0)
clk(1)
clk

FIG. 17

FIG. 18


c0
c1
c2
c3
c4
c5
c6
c7
c8
c9
c10
clk
wtx(rtz)
wrx(rtx)
wdata(rdata)
d1
d2