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

(10) **DE 693 33 579 T2** 2005.07.28

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 0 605 028 B1**
(21) Deutsches Aktenzeichen: **693 33 579.3**
(96) Europäisches Aktenzeichen: **93 203 325.1**
(96) Europäischer Anmeldetag: **29.11.1993**
(97) Erstveröffentlichung durch das EPA: **06.07.1994**
(97) Veröffentlichungstag
der Patenterteilung beim EPA: **04.08.2004**
(47) Veröffentlichungstag im Patentblatt: **28.07.2005**

(51) Int Cl.[7]: **H04L 25/38**

(30) Unionspriorität:
9216021 **28.12.1992** **FR**

(73) Patentinhaber:
Hewlett-Packard Development Co., L.P., Houston, Tex., US

(74) Vertreter:
Schoppe, Zimmermann, Stöckeler & Zinkler, 82049 Pullach

(84) Benannte Vertragsstaaten:
DE, FR, GB

(72) Erfinder:
Thoulon, Pierre-Yves, F-38340 Voreppe, FR

(54) Bezeichnung: **Verfahren und Schaltung zur Erkennung der Nennfrequenz eines Logiksignals**

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

Gebiet der Erfindung

[0001] Die vorliegende Erfindung bezieht sich auf die Identifizierung der Nennfrequenz eines Logiksignals, von dem erwartet wird, dass dasselbe eine Frequenz aufweist, die im Wesentlichen gleich einer jeglichen einer Mehrzahl von diskreten Nennfrequenzen ist. Insbesondere aber nicht ausschließlich bezieht sich die vorliegende Erfindung auf die Erfassung der Sendegeschwindigkeit einer Vorrichtung, die mit einer seriellen Leitung verbunden ist.

Hintergrund der Technik

[0002] Bei modernen Datenverarbeitungssystemen, die Terminals umfassen, die durch serielle Leitungen mit einem Server verbunden sind, können die Seriell-Schnittstellenschaltungen des Servers, allgemein als universeller asynchroner Empfänger-Sender (UART = Universal Asynchronous Transmitter) bezeichnet, automatisch an die verschiedenen möglichen Sendegeschwindigkeiten der Terminals angepasst werden. Auf eine anfängliche Verbindung des Terminals mit einem Server hin drückt ein Benutzer eine spezifische Taste des Terminals, allgemein die „Wagen-Rückgabe"-(CR- = carrige return) Taste, einmal oder mehrmals, um zu ermöglichen, dass der Server die Sendegeschwindigkeit erfasst und die zugeordnete UART-Schaltung desselben anpasst oder reprogrammiert."

[0003] Fig. 1 stellt eine Reihe von Zuständen einer seriellen Leitung während des Sendens eines „CR"-Schriftzeichens dar, das gemäß einem ASCII-Standard codiert ist. Anfänglich befindet sich in einem Bereitschaftszustand die serielle Leitung bei 1. Bei dem Beginn der Sendung des Schriftzeichens wird die serielle Leitung während der Sendezeit eines Bits 0. Dieses anfängliche Bit bei 0 wird als ein START-Bit bezeichnet und gibt systematisch den Beginn einer Sendung an. Dann treten sieben Schriftzeichencodebits in ansteigender Gewichtung auf. Für ein „CR"-Schriftzeichen sind diese Bits 1011000. Das achte Bit P, das nach dem START-Bit gesendet wird, ist gemäß dem Sendeprotokoll entweder ein achtes höchstwertiges Bit des gesendeten Schriftzeichens (für die „CR"-Taste 0) oder ein Paritätsbit, das zu 0 oder 1 gesetzt ist, so dass die Anzahl von gesetzten Bits bei 1 gerade ist (dieses Bit ist für die „CR"-Taste 1). Das letzte gesendete Bit, STOP, ist systematisch bei 1 und signalisiert das Ende einer Sendung.

[0004] Die Intervalle bei 1 oder bei 0, die jedem gesendeten Bit zugeordnet sind, haben die gleiche Länge, deren Wert von der Sendegeschwindigkeit abhängt. Für eine herkömmliche serielle Sendung ist eine Sendegeschwindigkeit aus einer Mehrzahl von Standardgeschwindigkeiten gewählt, die zum Beispiel 75 Bit/Sekunde und doppelte Geschwindigkeiten bis zu 38400 Bit/Sekunde, oder 57600 Bit/Sekunde und doppelte Geschwindigkeiten bis zu beispielsweise 921600 Bit/Sekunde sind. Die Geschwindigkeiten, die 57600 Bit/Sekunde überschreiten, oder die Geschwindigkeiten unter 38400 Bit/Sekunde weisen eine geometrische Fortschreitung bzw. Progression (mit einem Multiplikationsfaktor 2) auf. Jede Geschwindigkeit ist eine Nenngeschwindigkeit, die einer Nennzeitdauer der gesendeten Bits entspricht. Die wirksame Zeitdauer eines Bits kann innerhalb eines Bereichs von bis zu ±33% der Nennzeitdauer variieren, während die Nenngeschwindigkeitsvariation 2% nicht überschreitet.

[0005] Ein herkömmliches Verfahren zum Erfassen einer Sendegeschwindigkeit besteht in einem Messen der Breite des START-Bits bei dem Server, einem Reprogrammieren des UART auf die erfasste Geschwindigkeit während der Sendung des zweiten Bits, einem Erfassen der acht folgenden Bits (011000, P, STOP) und einem Überprüfen, ob diese Bits die Richtigen sind. Ein Fehler während dieser Verifizierung gibt entweder an, dass das gesendete Schriftzeichen nicht korrekt war (nicht die „CR"-Taste war) oder dass die Messung des START-Bits fehlschlug (zum Beispiel weil dieses Bit verzerrt war oder eine Messung zu spät startete). In diesem Fall wartet das System auf das Erteilen eines neuen Schriftzeichens „CR" auf der Leitung.

[0006] Um die Länge eines Zeitintervalls (oder eines Bits) zu bestimmen, wird herkömmlicherweise die Anzahl von Pulsen eines festen Frequenztakts gezählt, der innerhalb des Zeitintervalls auftritt, das gemessen werden soll. Die gezählte Anzahl von Pulsen liefert eine direkte Angabe der Intervalllänge. Zum Identifizieren einer Nennfrequenz unter mehreren möglichen Nennfrequenzen jedoch kann diese gezählte Anzahl nicht direkt verwendet werden, weil es ferner notwendig ist, die Nennfrequenz zu bestimmen, der diese Anzahl entspricht. Wenn die Anzahl einmal gezählt wurde, muss dieselbe zu diesem Zweck aufeinander folgend mit einer Mehrzahl von Minimal- und Maximalwerten verglichen werden, wobei jedes Paar von Minimal- und Maximalwerten einem Fehlerbereich um einen Nennwert entspricht.

[0007] Ein Nachteil eines derartigen Verfahrens ist, dass zwei Grenzwerte für jede mögliche Nennfrequenz gespeichert werden müssen, was zweimal so viele Speicherzellen wie Nennfrequenzen erfordert. Falls die Anzahl von Nennfrequenzen groß ist, sind zusätzlich die Grenzwerte, die der niedrigsten Nennfrequenz zugeordnet sind, hoch und erfordern eine Speicherzelle großer Größe. Falls zum Beispiel 16 Taktpulse in einem Bit bei der Geschwindigkeit von 38400 Bit/Sekunde gezählt werden, dann werden in der Tat 4096 Taktpulse in einem Bit bei der Geschwindigkeit von 75 Bit/Sekunde gezählt. Die An-

zahl 4096 erfordert 12 Bits, die gespeichert werden sollen, während die Anzahl 16 lediglich 4 Bit benötigt.

[0008] Die einfachste und kostengünstigste herkömmliche Weise eines Implementierens des obigen Verfahrens besteht darin, die Schritte desselben unter Verwendung einer Software auszuführen, wobei diese Software im Allgemeinen durch einen Mikroprozessor des Servers ausgeführt wird, der die seriellen Verbindungen überwacht. Die begrenzte Ausführungsgeschwindigkeit des Programms ermöglicht jedoch nicht die Erfassung von hohen Geschwindigkeiten und/oder ermöglicht nicht, dass eine geeignete Bestimmung der Sendegeschwindigkeiten vorgenommen wird, falls eine große Anzahl von seriellen Verbindungen simultan verwendet wird. Falls erwünscht ist, eine große Anzahl von seriellen Verbindungen angemessen zu verwalten und/oder die Geschwindigkeit dieser seriellen Verbindungen zu erhöhen, während dieser Ansatz verwendet wird, muss die Anzahl von Mikroprozessoren, die den Verbindungen zugewiesen sind, erheblich erhöht werden, was teuer ist.

Zusammenfassung der Erfindung

[0009] Eine Aufgabe der Erfindung besteht darin, ein Verfahren zum Identifizieren der Nennfrequenz eines Signals zu schaffen, dessen Frequenz im Wesentlichen gleich einer von einer Mehrzahl von Nennfrequenzen sein kann, wobei das Verfahren insbesondere einfach in einer Hardware zu implementieren ist.

[0010] Eine andere Aufgabe der Erfindung besteht darin, eine Schaltung zu schaffen, die das oben beschriebene Verfahren implementiert und zu einem schnellen Identifizieren der Nennfrequenz des Signals in der Lage ist.

[0011] Eine weitere Aufgabe der Erfindung besteht darin, eine Sende-/Empfangsschaltung zu schaffen, die für die Sendegeschwindigkeit einer seriellen Leitung selbstanpassend ist und zu einem Verarbeiten von insbesondere hohen Sendegeschwindigkeiten in der Lage ist.

[0012] Um diese Aufgaben zu lösen, betrifft die Erfindung die aufeinander folgende Sendung von Folgen von Taktpulsen beginnend von einer ersten vorbestimmten Flanke (zum Beispiel einer abfallenden Flanke) eines Signals, das gemessen werden soll, und für ein Zeitintervall andauernd, das zwischen der ersten Flanke und einer zweiten vorbestimmten Flanke des Signals, das gemessen werden soll, gebildet ist, wobei die Folgen von Taktpulsen jeweils eine jeweilige vorbestimmte Zeitdauer aufweisen, wobei die erste Folge bei einer Frequenz ist, die der höchstmöglichen Nennfrequenz entspricht, und die letzte Folge der niedrigsten Nennfrequenz entspricht. Die Zeitdauern dieser Folgen werden zuvor berechnet, so dass, falls das Signal, das gemessen werden soll, bei der n-ten Nennfrequenz von der höchsten Nennfrequenz ist, es die n-te Folge von Taktpulsen ist, die gesendet wird, wenn die zweite Flanke auftritt. Natürlich wird die Zeitdauer für jede Taktpulsfolge berechnet, um die mögliche Periodenvariation des Signals, das gemessen werden soll, um jede Nennperiode zu berücksichtigen.

[0013] Formeller gesagt ist gemäß einem Aspekt der vorliegenden Erfindung ein Verfahren zum Identifizieren der Nennfrequenz eines Logiksignals bereitgestellt, das eine Frequenz aufweisen kann, die, mit einem möglichen Fehler, gleich einer von n diskreten Frequenzen mit sich fortlaufend verringerndem Wert von der ersten bis zu der n-ten Frequenz ist, wobei der mögliche Fehler, für jede der diskreten Frequenzen, einen vorbestimmten zulässigen Bereich aufweist, der durch einen maximalen positiven und einen maximalen negativen Fehlerwert definiert ist, wobei das Verfahren während einer Messungsperiode implementiert wird, die durch zwei vorbestimmte Flanken des Logiksignals (Rx) eingerichtet ist, und das Verfahren die folgenden Schritte umfasst:

(a) zu dem Ausmaß, das durch die Dauer der Messungsperiode gestattet ist, sequentielles Senden von Folgen von Taktpulsen, wobei jede aufeinanderfolgende Taktpulsfolge eine geringere Frequenz als die Vorgängerin derselben aufweist und eine jeweilige vorbestimmte Zeitdauer aufweist, wobei die vorbestimmten Zeitdauern der Taktpulsfolgen derart sind, dass:

(i) die Gesamtdauer der Pulsfolgen von der ersten zu der i-ten Pulsfolge, für $0 < i < n$, zumindest so groß wie die Dauer der Messungsperiode ist, die in dem Fall resultieren würde, dass das Logiksignal die i-ten diskreten Frequenzen aufweist, die durch den entsprechenden maximalen negativen Fehlerwert verringert sind, aber geringer als die Dauer der Messungsperiode ist, die in dem Fall resultieren würde, dass das Logiksignal die (i + i)-te diskrete Frequenz aufweist, die durch den entsprechenden maximalen positiven Fehlerwert erhöht ist,

(ii) die Gesamtdauer der Pulsfolgen von der ersten zu der n-ten Pulsfolge zumindest so groß wie die Dauer der Messungsperiode ist, die in dem Fall resultieren würde, dass das Logiksignal die n-te diskrete Frequenz aufweist, die durch den entsprechenden maximalen positiven Fehlerwert verringert ist,

(b) bei dem Auftreten der zweiten der vorbestimmten Flanken, Nehmen einer Angabe der dann aktuellen Taktfrequenz als eine Angabe der Nennfrequenz des Logiksignals.

[0014] Vorzugsweise sind die Zeitdauern der Taktpulsfolgen derart, dass die Summe der Zeitdauern der ersten bis zu der i-ten Folge von Taktpulsfolgen,

für 0 < i ≤ n, im Wesentlichen gleich der Dauer der Messungsperiode ist, die in dem Fall resultieren würde, dass das Logiksignal eine Frequenz aufweist, die gleich der i-ten diskreten Frequenz ist, die durch den entsprechenden maximalen negativen Fehlerwert verringert ist.

[0015] Vorteilhafterweise umfasst das Verfahren ferner den Schritt eines Angebens, dass die Nennfrequenz des Logiksignals unbekannt ist, falls die zweite der Flanken vor dem Ende der Sendung der n-ten Taktpulsfolge nicht auftritt. Alternativ oder zusätzlich kann das Verfahren den Schritt eines Angebens umfassen, dass die Nennfrequenz des Logiksignals unbekannt ist, falls, wenn die zweite der Flanken während der i-ten Pulsfolge für 0 < i ≤ n auftritt, die Dauer der tatsächlichen Messungsperiode geringer als die Dauer der Messungsperiode ist, die in dem Fall resultieren würde, dass das Logiksignal eine Frequenz aufweist, die gleich der i-ten diskreten Frequenz ist, die durch den entsprechenden maximalen positiven Fehlerwert erhöht ist.

[0016] Vorzugsweise ist das Verhältnis des Werts der i-ten diskreten Frequenz zu dem Frequenzwert der i-ten Taktpulsfolge für alle Werte von i für 0 < i ≤ n das gleiche.

[0017] Gemäß einem anderen Aspekt der vorliegenden Erfindung ist eine Frequenzidentifizierungsschaltung zum Identifizieren der augenblicklichen Nennfrequenz eines Logiksignals bereitgestellt, das eine augenblickliche Frequenz aufweisen kann, die mit einem möglichen Fehler, gleich einer von einer Mehrzahl von diskreten Nennfrequenzen ist, wobei die Schaltung folgende Merkmale aufweist:

– einen Taktgenerator zum Liefern einer Mehrzahl von Taktsignalen, wobei jedes Taktsignal eine Frequenz aufweist, die ein konstantes Mehrfaches einer zugeordneten der diskreten Frequenz ist;

– einen Sequenzierer zum Liefern eines Startsignals bei dem Auftreten einer ersten vorbestimmten Flanke des Logiksignals und eines Stoppsignals bei dem Auftreten einer zweiten vorbestimmten Flanke des Logiksignals;

– einen ersten Zähler zum Enthalten von Auswahldaten, deren aufeinanderfolgende Werte jeweiligen Taktsignalen von sich fortlaufend verringernden Frequenzen und somit der jeweiligen diskreten Frequenz mit sich fortlaufend verringerndem Wert zugeordnet sind;

– einen Multiplexer, der mit dem Taktgenerator verbunden ist, und durch die Auswahldaten gesteuert ist, um das zugeordnete Taktsignal zu liefern;

– einen zweiten Zähler zum Empfangen des Ausgangs des Multiplexers an einem Zähleingang;

– einen Komparator zum Liefern eines Freigabesignals, wenn der Inhalt des zweiten Zählers einen Maximalwert erreicht, der in Abhängigkeit von den Auswahldaten ausgewählt ist; und

– eine Steuerschaltung zum Liefern von Initialisierungssignalen zu dem ersten und dem zweiten Zähler, sobald dieselbe das Startsignal empfängt, und zum Liefern eines Zählfreigabesignals zu dem ersten Zähler und eines Initialisierungssignals zu dem zweiten Zähler, wenn das Freigabesignal durch den Komparator geliefert wird, wobei die Steuerschaltung in einen Bereitschaftszustand geht, wenn dieselbe das Stopsignal empfängt.

[0018] Vorzugsweise entspricht der Maximalwert, der in Abhängigkeit von den Auswahldaten ausgewählt ist, dem Wert, der durch den zweiten Zähler bei dem Auftreten der zweiten vorbestimmten Flanke des Logiksignals in dem Fall erreicht worden wäre, dass letzteres die diskrete Frequenz aufweist, die den vorliegenden Auswahldaten zugeordnet ist, verringert durch einen maximalen zulässigen Fehler.

[0019] Vorteilhafterweise liefert der Komparator ein Unidentifizierte-Frequenz-Signal (REJ), das angibt, dass die Frequenz des Logiksignals nicht einer diskreten Nennfrequenz entspricht, wenn eines der folgenden auftritt:

– die Auswahldaten erreichen einen Wert jenseits dem, der der niedrigsten diskreten Nennfrequenz zugeordnet ist,

– der Inhalt des zweiten Zählers ist niedriger als ein Minimalwert, der dem Wert entspricht, der durch den zweiten Zähler bei dem Auftreten der zweiten vorbestimmten Flanke des Logiksignals in dem Fall erreicht worden wäre, dass letzteres die diskrete Frequenz aufweist, die den vorliegenden Auswahldaten zugeordnet ist, erhöht durch einen maximalen zulässigen Fehler.

[0020] Vorzugsweise sind der Maximal- und der Minimalwert, die den Werten der Auswahldaten entsprechen, in jeweiligen Speichern gespeichert, die durch die Auswahldaten adressiert werden, wobei der Wert, der durch jeden Speicher geliefert wird, mit dem Ausgang des zweiten Zählers in einem jeweiligen Komparator verglichen wird.

[0021] In noch einem anderen Aspekt stellt die vorliegende Erfindung eine Schaltung bereit, die sich selbst an die Sendegeschwindigkeit einer Vorrichtung anpasst, die mit einer seriellen Leitung verbunden ist, auf der die Vorrichtung ein vorbestimmtes Schriftzeichen für die Erfassung der Sendegeschwindigkeit sendet, die folgende Merkmale umfasst:

– die Frequenzidentifizierungsschaltung wie oben dargelegt;

– einen universellen asynchronen Empfänger-Sender (UART = Universal Asynchronous Receiver Transmitter), der mit der seriellen Leitung verbunden ist und durch das Taktsignal getaktet

ist, das durch den Multiplexer der Identifizierungsschaltung geliefert wird; und
– eine Überprüfungsschaltung zum Liefern eines Korrektes-Schriftzeichen-Signals, falls das Schriftzeichen, das durch den UART empfangen wird, gleich dem vorbestimmten Schriftzeichen ist;

wobei der Sequenzierer der Frequenzidentifizierungsschaltung mit der seriellen Leitung zum Liefern des Startsignals bei dem Auftreten einer ersten Flanke an der seriellen Leitung, die dem ersten Bit des vorbestimmten Schriftzeichen entspricht, und zum Liefern des Stoppsignals bei dem Auftreten einer zweiten Flanke an der seriellen Leitung, die einem vorbestimmten nachfolgenden Bit des vorbestimmten Schriftzeichens entspricht, verbunden ist, wobei der Sequenzierer ferner wirksam ist, um ein weiteres Startsignal zu dem UART zu liefern, wenn das Unidentifizierte-Frequenz-Signal nach dem Auftreten des Stop-Signals nicht vorhanden ist, und um das Taktsignal, das zu dem UART geliefert wird, unverändert beizubehalten, wenn das Korrektes-Schriftzeichen-Signal durch die Überprüfungsschaltung am Ende der Sendung des vorbestimmten Schriftzeichens geliefert wird.

[0022] Ein Vorteil der vorliegenden Erfindung, inhärent in dem Verfahren gemäß der Erfindung, besteht darin, dass dasselbe unter Verwendung von Logikschaltungen besonders einfach zu implementieren ist.

[0023] Ein anderer Vorteil, der mit Ausführungsbeispielen der vorliegenden Erfindung erzielbar ist, besteht darin, dass die Minimal- und Maximalwerte, die jeder Nennfrequenz zugeordnet sind, im Wesentlichen gleich sind und eine konstante Speicherzelle kleiner Größe für die Speicherung derselben erfordern. Falls die aufeinander folgenden Nennfrequenzen in einer geometrischen Fortschreitung bzw. Progression sind, sind die Minimal- oder Maximalwerte beginnend von der zweiten Folge von Taktpulsen gleich.

[0024] Ein weiterer Vorteil, der mit speziellen Ausführungsbeispielen der vorliegenden Erfindung erreicht werden kann, besteht darin, dass die Taktfrequenz direkt durch einen UART verwendet werden kann, der mit der Nennfrequenz des Signals, das gemessen werden soll, synchronisiert sein muss.

[0025] Ein weiterer Vorteil der vorliegenden Erfindung besteht darin, dass die Identifizierung der erwünschten Nennfrequenz allgemein genau in dem Moment ausgeführt werden kann, wenn die zweite Flanke auftritt und nicht nach nachfolgenden Vergleichsoperationen wie bei einem herkömmlichen System; dies ermöglicht, dass die Zeit, die für die Frequenzidentifizierung benötigt wird, verringert wird.

Kurze Beschreibung der Zeichnungen

[0026] Ein Frequenzidentifizierungsverfahren gemäß der Erfindung und eine Frequenzidentifizierungsschaltung, die die Erfindung ausführt, werden nun insbesondere durch ein nicht einschränkendes Beispiel mit Bezug auf die zugehörigen schematischen Zeichnungen beschrieben, in denen:

[0027] Fig. 1 oben beschrieben den Signalverlauf eines Seriell-Sendesignals eines spezifischen Schriftzeichens darstellt, das herkömmlicherweise gesendet wird, um zu ermöglichen, dass eine Empfangsschaltung die Sendegeschwindigkeit bestimmt;

[0028] Fig. 2 ein exemplarisches Taktsignal während der Ausführung des Verfahrens gemäß der Erfindung zum Identifizieren der Nennfrequenz eines Logiksignals darstellt;

[0029] Fig. 3 ein Blockdiagramm eines ersten Ausführungsbeispiels der Frequenzidentifizierungsschaltung ist, die die Erfindung ausführt;

[0030] Fig. 4 ein Flussdiagramm von Operationen ist, die durch eine Steuerschaltung der Schaltung von Fig. 3 ausgeführt werden;

[0031] Fig. 5 Elemente eines der Blöcke der Schaltung von Fig. 3 detaillierter darstellt;

[0032] Fig. 6 ein Blockdiagramm einer Empfänger-/Senderschaltung ist, die die Erfindung ausführt und für die Sendegeschwindigkeit einer seriellen Leitung selbstanpassend ist; und

[0033] Fig. 7 ein Flussdiagramm von Operationen ist, die durch einen Sequenzierer der Schaltung von Fig. 6 ausgeführt werden.

Bester Modus zum Ausführen der Erfindung

[0034] Fig. 2 stellt exemplarische Schritte des Frequenzidentifizierungsverfahrens, das beschrieben werden soll, für die Identifizierung der Nennfrequenz eines Logiksignals Rx dar, das eine jegliche einer Mehrzahl von diskreten Nennfrequenzen aufweisen kann. In Fig. 2 entsprechen derartige diskrete Nennfrequenzen Standardsendegeschwindigkeiten auf einer seriellen Leitung bis zu 921600 Bit/Sekunde.

[0035] In der folgenden Beschreibung werden Ausdrücke, wie beispielsweise „Nenngeschwindigkeit", „Standardgeschwindigkeit" und „Nennfrequenz" austauschbar verwendet. Da das Verfahren, das beschrieben werden soll, auf lediglich eine oder mehrere Halbperioden eines Signals angewendet wird, bezeichnet der Ausdruck „Frequenz" eine momentane Frequenz präziser, die aus den betrachteten Halbperioden berechnet wird.

[0036] Das Identifizierungsverfahren wird bei einem Zeitpunkt t0 initialisiert, der dem Beginn eines spezifischen Messintervalls eines Signals Rx entspricht, dessen Frequenz identifiziert werden soll. Dieses Messintervall ist durch zwei vorbestimmte Flanken definiert, die einer oder mehreren Halbperioden Rx entsprechen. In Fig. 2 wird angenommen, dass das Messintervall den ersten zwei Bits (START und 1) des Sendesignals eines Identifizierungsschriftzeichens entspricht, wie beispielsweise des CR-Schriftzeichens.

[0037] Zwischen dem Zeitpunkt t_0 und nachfolgenden, aufeinander folgenden Zeitpunkten t_1, t_2 ... t_i ... t_n werden Folgen von Taktpulsen CK1, CK2 ... CK_i ... CK_n gesendet. Taktpulse CK sind in Fig. 2 durch vertikale Linien angegeben. Jede Pulsfolge ist einer der Nenngeschwindigkeiten zugeordnet. Die erste Pulsfolge, CK1, ist der maximalen ausgewählten Geschwindigkeit (921600 Bit/Sekunde) zugeordnet und die letzte Pulsfolge, CK_n, ist der niedrigsten Geschwindigkeit zugeordnet, zum Beispiel 75 Bit/Sekunde.

[0038] Die Verhältnisse zwischen jeder Nennfrequenz und der Frequenz der zugeordneten Pulsfolge sind gleich. Wie es in Fig. 2 gezeigt ist, verringern sich somit die Frequenzen der aufeinander folgenden Pulsfolgen um das gleiche Verhältnis wie die aufeinander folgenden Standardgeschwindigkeiten.

[0039] Die Zeitpunkte t_1, t_2 ... sind ausgewählt, so dass das Ende des Messintervalls eines Signals Rxi bei der i-ten Nennfrequenz (i = 1, 2 ...) bereitgestellt ist, während die i-te Pulsfolge gesendet wird. Ganz am Ende des Messintervalls wird der Rang der Pulsfolge, die gesendet wird, als eine Angabe der erwünschten Nennfrequenz genommen.

[0040] Fig. 2 zeigt durch ein Beispiel Signale Rx1, Rx4 und Rx7, die gemessen werden sollen. Die Messintervalle der Signale Rx1, Rx4 und Rx7 bei den jeweiligen Geschwindigkeiten von 921600, 115200 und 19200 Bit/Sekunde enden vor Zeitpunkten t_1, t_4 bzw. t_7. Die Pulsfolgen, die gesendet werden, sind die erste, die vierte bzw. die siebte Pulsfolge, was angibt, dass die erwünschten Nenngeschwindigkeiten die erste (921600), die vierte (115200) bzw. die siebte (119200) sind.

[0041] Die Zeitdauer eines Bits, das durch ein Signal Rxi gesendet wird, kann um den Nennwert desselben variieren; folglich darf das Messintervall den Nennwert desselben bis zu einem maximalen tolerierbaren Wert überschreiten, bevor die nächste Pulsfolge gesendet wird. Zeitpunkte t_i entsprechen exakt diesen Maximalwerten. Falls ein Messintervall einen Zeitpunkt t_i überschreitet, kann somit angenommen werden, dass die Geschwindigkeit des Signals, das gemessen werden soll, eine der Geschwindigkeiten mit einem höheren Rang als i ist oder dass das interessierende Signal ein fehlerhaftes Signal ist.

[0042] Der Fall eines derartigen fehlerhaften Signals ist durch ein Signal Rxe dargestellt. Das Messintervall des Signals Rxe endet zwischen Zeitpunkten t4 und t5. Die Nenngeschwindigkeit desselben wäre als die fünfte (57600) Geschwindigkeit angegeben. Wie es in Fig. 2 zu ersehen ist, unterscheidet sich die Geschwindigkeit des Signals Rxe jedoch erheblich von 57600 und befindet sich ohnehin außerhalb eines tolerierbaren Fehlerbereichs von in etwa 57600 Bit/Sekunde. Schraffierte Bereiche stellen Bereiche eines tolerierbaren Fehlers um jede der Nenngeschwindigkeiten dar. Der Bereich, der der i-ten Geschwindigkeit zugeordnet ist, endet bei dem Zeitpunkt t_i und beginnt bei einem Zeitpunkt t_i'. Der Vereinfachung halber sind Referenzen t_i' und t_2' in Fig. 2 nicht gezeigt. Der Fehlerbereich, der der i-ten Geschwindigkeit zugeordnet ist, entspricht im Wesentlichen einer Variation um die i-te Geschwindigkeit um ±25% der Differenz, die die i-te Geschwindigkeit von der i – 1-ten Geschwindigkeit trennt.

[0043] Ein anderer Schritt des Frequenzidentifizierungsverfahrens besteht in einem Erfassen, zu der Zeit, zu der ein Messintervall endet, ob das Signal innerhalb eines der tolerierbaren Fehlerbereiche ist. Dieses Erfordernis wird für die Signale Rx1, Rx2 und Rx7 erfüllt, aber wird für das Signal Rxe nicht erfüllt, dessen Messintervall zwischen dem Zeitpunkt t_4 und t_5' endet. Im Folgenden ist hierin detaillierter zu ersehen, wie erfasst werden soll, ob ein Signal in einen Fehlerbereich fällt oder nicht.

[0044] Falls das Messintervall nicht vor dem Zeitpunkt t_n endet, der der niedrigstmöglichen Nenngeschwindigkeit zugeordnet ist, wird zusätzlich die Geschwindigkeit des gemessenen Signals als unbekannt angegeben.

[0045] Das Frequenzidentifizierungsverfahren kann auf eine einfache Weise durch ein Verwenden eines Zählers implementiert werden, der für jede Taktpulsfolge die Taktpulse CK_i zählt, das heißt ein Zähler, bei jedem von Malen t_i – 1 rückgesetzt wird. Die Werte, die durch den Zähler gezählt werden, sind in Fig. 2 mit L etikettiert.

[0046] Bei dem Beispiel von Fig. 2 wird angenommen, dass die Taktfrequenz, die der i-ten Nenngeschwindigkeit zugeordnet ist, derart ist, dass das Messintervall eines Signals bei der i-ten Nenngeschwindigkeit gleich 16 mal der Taktperiode ist. Für Folgen von Taktpulsen CK2–CK5 und CK7 und darüber, deren Frequenz zweimal die unmittelbar niedrigere Frequenz ist, entspricht in diesem Fall eine Nennmessdauer, das heißt die Messdauer eines Logiksignals bei einer der Nennfrequenzen, dem Zählen von 7 Taktpulsen. Die minimale Messdauer, das

heißt die Nenndauer mit einem maximalen negativen Fehler, entspricht dem Zählen von 5 Taktpulsen. Die maximale Messdauer, das heißt die Nenndauer mit einem maximalen positiven Fehler, entspricht dem Zählen von 9 Taktpulsen.

[0047] Für die erste Taktpulsfolge CK1 entspricht die Nennmessdauer dem Zählen von 16 Taktpulsen; die minimale Messdauer entspricht dem Zählen von 14 Taktpulsen und die maximale Dauer entspricht dem Zählen von 18 Taktpulsen.

[0048] Für die sechste Taktpulsfolge CK6, deren Frequenz nicht die Hälfte der Frequenz der vorhergehenden Pulsfolge CK5 ist, entspricht die Nennmessdauer dem Zählen von 4 Taktpulsen, die maximale Messdauer entspricht dem Zählen von 6 Taktpulsen und die minimale Messdauer dem Zählen von 3 Taktpulsen.

[0049] Um die Gültigkeit der Nenngeschwindigkeit zu überprüfen, die am Ende eines Messintervalls angegeben wird, wird der Zustand des Zählers gelesen und eine Überprüfung wird vorgenommen, dahingehend ob der Letztere höher als der Minimalwert ist, der der Nenngeschwindigkeit zugeordnet ist, die angegeben ist. Für das Signal Rxe ist der Zustand des Zählers am Ende des Messintervalls 4, während derselbe höher als 5 sein sollte; diese Situation gibt an, dass die Geschwindigkeit des Signals Rxe unbekannt ist und deshalb dass die angegebene Geschwindigkeit nicht gültig ist.

[0050] Fig. 3 ist ein Blockdiagramm einer Schaltung, die einen Zähler einsetzt, der auf die oben erwähnte Weise verwendet wird, um das Frequenzidentifizierungsverfahren zu implementieren. Ein Sequenzierer (SEQ) **10** empfängt ein Signal Rx, dessen Nennfrequenz identifiziert werden soll. Der Sequenzierer **10** liefert einen Startpuls STARTC und einen Stopppuls STOPC zu einer Steuerschaltung (CTRL) **12** auf das Auftreten einer ersten Flanke beziehungsweise einer zweiten Flanke des Signals Rx hin. Die erste und die zweite Flanke sind zum Beispiel zwei aufeinander folgende Flanken, die die gleiche Richtung aufweisen, wobei eine Periode des Signals Rx oder ein Messintervall bestimmt ist. Die zwei Flanken könnten ferner zwei aufeinander folgende Flanken entgegengesetzter Richtung sein, wobei eine Halbperiode bestimmt ist, oder zwei Flanken, die gleiche Richtung aufweisen und voneinander durch mehrere Perioden getrennt sind. Ein exemplarischer höher entwickelter Sequenzierer ist hierin im Folgenden detaillierter beschrieben.

[0051] Die Steuerschaltung **12** liefert ein Initialisierungssignal INITBR und ein Dekrementssignal DECBR zu einem Zähler (COUNT BR) **14**, dessen Inhalt, hierin im Folgenden als Auswahldaten BR bezeichnet, einen zugeordneten Kanal eines Multiplexers **16** auswählt.

[0052] Ein Taktgenerator (CLK) **18** liefert eine Mehrzahl von Taktsignalen, die verschiedene Frequenzen aufweisen, zu dem Multiplexer (MUX) **16**. Jedes Taktsignal entspricht einer der Pulsfolgen, die mit Bezug auf Fig. 2 erwähnt sind, und ist deshalb einer der möglichen Nennfrequenzen des Signals Rx zugeordnet. Somit geben die Daten BR die Nennfrequenz an, die dem Taktsignal zugeordnet ist, das durch den Multiplexer ausgewählt ist. Die Auswahldaten BR werden bei dem Maximalwert derselben initialisiert, wenn das Signal INITBR gesetzt wird. Die abnehmenden Werte der Daten BR wählen in dem Multiplexer **16** Taktsignale, die abnehmende Frequenzen aufweisen, aus dem Taktgenerator **18** aus. Das ausgewählte Taktsignal CK wird zu dem Zähleingang des Zählers **20** geliefert, der von der Steuerschaltung **12** ein Rücksetzsignal RSTL und ein Freigabesignal ENL empfängt (wenn das Signal ENL gesetzt wird, wird der Inhalt des Zählers **20** zu dem Ausgang L desselben übertragen). Der Zähler (COUNT L) **20** entspricht dem Zähler, der mit Bezug auf Fig. 2 erwähnt ist.

[0053] Eine Verifiziererschaltung (VERIF) **22** empfängt den Ausgang L des Zählers **20** und die Auswahldaten BR. Die Schaltung **22**, die hierin im Folgenden detaillierter beschrieben ist, ist angeordnet, um ein Signal L = MAX zu der Steuerschaltung **12** zu liefern, wenn der Ausgang L des Zählers **20** gleich einem Maximalwert ist, der eine Funktion der Daten BR (und somit der zugeordneten Nennfrequenz) ist. Die Maximalwerte entsprechen den Zuständen des Zählers **20** bei den Zeitpunkten t_1, t_2 ... von Fig. 2. Die Schaltung **22** liefert ein Signal BR = 0 zu der Steuerschaltung **12**, wenn die Daten BR einen Wert, zum Beispiel 0, erreichen, der einem nicht verfügbaren Taktsignal entspricht, das eine zu niedrige Frequenz aufweist. Schließlich liefert die Schaltung **22** ein Unterdrückungssignal REJ, wenn der Ausgang L des Zählers **20** niedriger als ein Minimalwert ist, der ebenfalls eine Funktion der Daten BR (und deshalb der zugeordneten Frequenz) ist. Die Minimalwerte entsprechen den Zuständen des Zählers **20** zu Zeitpunkten t_1', t_2' ... von Fig. 2.

[0054] Unter erneuter Bezugnahme auf Fig. 2 ist die Schaltung von Fig. 3 schematisch wie folgt wirksam. Zu dem Zeitpunkt t_0 liefert der Sequenzierer **10** einen Puls STARTC zu der Steuerschaltung **12**, die daraufhin den Zähler **14** (die Daten BR) zu dem höheren Wert desselben initialisiert und den Zähler **20** rücksetzt. Die Daten BR bei einem Maximalwert wählen das Taktsignal CK aus, das die höchste Frequenz aufweist. Der Zähler **20** zählt Taktpulse, beginnend von Null. Wenn der Zustand L des Zählers **20** einen Maximalwert erreicht, der durch die Daten BR (einen der Werte zu den Zeitpunkten t_1, t_2 ...) ausgewählt ist, liefert die Verifiziererschaltung **22** einen Puls L =

MAX zu der Steuerschaltung **12**, die die Daten BR dekrementiert und den Zähler **20** rücksetzt. Die neuen Daten BR wählen einen neuen Maximalwert in der Verifiziererschaltung **22** aus, die wiederum darauf wartet, dass der Zustand des Zählers **20** diesen neuen Maximalwert erreicht. Auf das Auftreten der zweiten fallenden Flanke des Signals Rx (das Ende des Messintervalls) hin liefert der Sequenzierer einen Puls STOPC zu der Steuerschaltung **12**, die den Zustand des Zählers **20** einfriert. Der Zustand L wird mit einem Minimalwert (einem der Werte zu den Zeitpunkten t'_1, t'_2 ...) verglichen, der durch die Daten BR in der Verifiziererschaltung **22** ausgewählt ist, die das Unterdrückungssignal REJ setzt, falls der Wert L niedriger als dieser Minimalwert ist.

[0055] Fig. 4 ist ein Flussdiagramm der Operationen, die durch die Steuerschaltung **12** ausgeführt werden, und erleichtert ein Verständnis der detaillierten Operation der Schaltung von Fig. 3. Hierin geht im Folgenden einem Absatz, der einen Block des Flussdiagramms beschreibt, die Blocknummer voran.

[0056] **100**. Die Steuerschaltung **12** führt eine Schleife aus, während dieselbe auf die Aktivierung des Signals STARTC wartet.

[0057] **102**. Das Signal STARTC wird gesetzt (zum Beispiel zu dem Zeitpunkt t_0). Die Signale INITBR, RSTL und ENL werden simultan zum Initialisieren der Auswahldaten BR bei dem Maximalwert derselben zum Rücksetzen des Inhalts des Zählers **20** bzw. Übertragen des Inhalts L zu dem Eingang der Verifiziererschaltung **22** gesetzt. Die Initialisierung der Daten BR bei dem Maximalwert derselben bewirkt die Auswahl des Taktsignals CK, das eine maximale Frequenz aufweist.

[0058] **104**. Eine Überprüfung wird vorgenommen, ob das Signal STOPC gesetzt ist. Falls ja, ist der Frequenzidentifizierungsprozess vollständig und das System kehrt zu dem Wartezustand von Block **100** zurück.

[0059] **106**. Das Signal STOPC ist nicht gesetzt. Eine Überprüfung wird vorgenommen, ob das Signal L = MAX gesetzt ist. Der Zweck dessen besteht darin, herauszufinden, ob zum Beispiel einer der Zeitpunkte t_1, t_2 ... erreicht wurde.

[0060] **108**. Das Signal L = MAX ist nicht gesetzt. Das Signal ENL wird gesetzt, um einen möglichen neuen Wert L des Zählers **20** auszugeben. Eine Ausführung wird bei dem Block **104** wieder aufgenommen.

[0061] Die Schleife **104**-**106**-**108** wird wiederholt, während der Zähler **20** simultan bei der Rate des ausgewählten Taktsignals CK inkrementiert wird.

[0062] **110**. Die Schleife **104**-**106**-**108** wird unterbrochen, wenn das Signal L = MAX durch die Verifiziererschaltung **22** gesetzt wird, was angibt, dass einer der Zeitpunkte t_1, t_2, ... erreicht wurde. Die Signale RSTL und DECBR werden gesetzt, um den Zähler **20** rückzusetzen bzw. um die Auswahldaten BR in dem Zähler **14** zu dekrementieren. Dann wählt der Multiplexer **16** eine neue Frequenz, die niedriger als die vorhergehende Frequenz ist, des Taktsignals CK aus und der Zähler **20** wird bei der Rate dieser neuen Frequenz inkrementiert.

[0063] **112**. Eine Überprüfung wird vorgenommen, ob ein Signal BR = 0 gesetzt ist. Das Signal BR = 0 wird gesetzt, wenn die Auswahldaten BR einen unteren Grenzwert, 0, erreichen, was angibt, dass die Frequenz des Signals Rx zu niedrig ist, um bestimmt zu werden. Falls ja, geht die Steuerschaltung **12** zu dem Wartezustand derselben bei Block **100**. Falls nicht, kehrt eine Ausführung zu der Schleife **104**-**106**-**108** zurück, wo der Inhalt L des Zählers **20** erneut inkrementiert wird.

[0064] Wenn das Setzen des Signals STOPC bei dem Block **104** einmal erfasst wird, tritt die Schaltung in einen inaktiven Wartemodus bei Block **100** ein. Der Ausgang L des Zählers **20** wird eingefroren. Die Daten BR werden ebenfalls eingefroren und geben die Nennfrequenz des Signals Rx an. Das Unterdrückungssignal REJ, das durch die Verifiziererschaltung **22** geliefert wird, gibt an, dass die Nennfrequenz unbekannt ist, falls der Wert L, der durch den Zähler **20** geliefert wird, den Minimalwert nicht erreicht hat, der den vorliegenden Daten BR zugeordnet ist, oder falls die Daten die untere Begrenzung derselben erreicht haben (bei Block **112**).

[0065] Fig. 5 stellt ein Ausführungsbeispiel der Verifiziererschaltung **22** dar, die an einen allgemeinen Fall angepasst ist. Die Auswahldaten BR werden zu den Adresseingängen von zwei ROMs **30** und **31** geliefert. Der ROM **30** liefert für jeden Wert der Daten BR einen minimalen zugeordneten Wert MIN zu einem Komparator (COMP) **34**. Der ROM **31** liefert für jeden Wert der Daten BR einen maximalen zugeordneten Wert MAX zu einem Komparator (COMP) **35**. Jeder Komparator **34** und **35** empfängt ferner den Wert L, der durch den Zähler **20** geliefert wird. Der Komparator **34** liefert ein Aktiv-Signal, wenn der Wert L niedriger als der Wert MIN ist, der durch den ROM **30** geliefert wird. Der Komparator **35** liefert das Signal L = MAX wenn der Wert L gleich dem Wert MAX ist, der durch den ROM **31** geliefert wird. Der Ausgang des Komparators **34** ist mit einem Eingang eines ODER-Gatters **37** verbunden, das das Fehlersignal REJ liefert. Jedes Bit der Daten BR wird ferner zu einem Eingang eines NOR-Gatters **39** geliefert, dessen Ausgang, der das Signal BR = 0 liefert, mit dem zweiten Eingang des Gatters **37** verbunden ist.

[0066] Bei dieser Konfiguration ist es möglich, jedem Wert der Daten BR, und deshalb jeder Nenngeschwindigkeit, eine minimale und eine maximale Begrenzung für einen tolerierbaren Fehlerbereich für das Messintervall zuzuordnen.

[0067] Die Taktfrequenzen sind gewählt, so dass die Verhältnisse zwischen jeder Nennfrequenz und der zugeordneten Taktfrequenz gleich sind. Ein Vorteil dieser Wahl von Taktfrequenzen ist, dass die Taktfrequenz, die bei dem Ende des Prozesses geliefert wird, wie hierin im Folgenden zu ersehen ist, dazu dienen kann, eine Schaltung direkt zu synchronisieren, die mit der erwünschten Nennfrequenz synchronisiert werden muss. Ein weiterer Vorteil dieser Wahl ist, dass alle Grenzwerte dieselbe Größenordnung aufweisen. Falls die Fortschreitung bzw. Progression der Nenngeschwindigkeiten geometrisch ist, sind alle Grenzwerte für Nenngeschwindigkeiten, die niedriger als der höchste Wert sind, ebenfalls gleich, wie es in Fig. 2 für Geschwindigkeiten von 460800 bis 57600 und für Geschwindigkeiten niedriger als 38400 gezeigt ist.

[0068] Bei dem Beispiel von Fig. 2 gibt es lediglich drei Minimalwerte, 14, 5 und 3, und drei entsprechende Maximalwerte 18, 9 und 6, die gespeichert werden sollen. Um eine derartige kleine Anzahl von Werten zu speichern, können die ROMs **30** und **31** mit einer Logikschaltung ersetzt werden, die die erwünschten Werte durch ein Decodieren der Daten BR liefert, wobei so das System vereinfacht wird.

[0069] Fig. 6 stellt eine Anwendung der Schaltung von Fig. 3 auf eine Empfänger/Sender-Schaltung dar, die für die Sendegeschwindigkeit einer seriellen Leitung selbstanpassend ist. Alle Elemente von Fig. 3 sind dargestellt und mit den gleichen Referenzen etikettiert. Die Identifizierungsschaltung von Fig. 3 ist einer UART-Schaltung **60** zugeordnet, die das Signal Rx empfängt und durch das Taktsignal CK getaktet ist, das durch den Multiplexer **16** geliefert wird. Der Sequenzierer **10** führt zusätzliche Operationen, die hierin im Folgenden beschrieben sind, mit Bezug auf den Sequenzierer von Fig. 3 aus. Der Sequenzierer **10** empfängt das Unterdrückungssignal REJ von der Verifiziererschaltung **22** und weist ein zugeordnetes Register (REG) **62** auf, auf das durch einen Mikroprozessorbus (nicht gezeigt) zugegriffen wird. Das Freigeben eines Bits START des Registers **62** gibt dem Sequenzierer **10** an, dass eine Anpassungssequenz gestartet werden soll. Am Ende der Anpassungssequenz liefert der Sequenzierer **10** ein Signal STORE zu dem Register **62**, wobei die Speicherung der Auswahldaten BR in dem Register bewirkt wird, und gibt ein Unterbrechungsbit IRQ des Registers frei, so dass der Mikroprozessor die Ergebnisse lesen und die gemessenen Charakteristika der Leitung Rx speichern kann.

[0070] Der UART **60** empfängt ein Signal CO von dem Sequenzierer **10** zum Starten der Erfassung eines Schriftzeichens auf der Leitung Rx. Wenn ein Schriftzeichen erfasst ist, gibt der UART ein Signal CAVAIL frei und liefert das Schriftzeichen auf Leitungen CHAR. Die Leitungen CHAR und das Signal CAVAIL werden zu einer Schriftzeichenprüfschaltung (VER CHAR) **64** geliefert, die dem Sequenzierer **10** ein Signal OK liefert, falls das Schriftzeichen, das auf den Leitungen CHAR vorliegt, einem vorgespeicherten Schriftzeichen entspricht. Die Schaltung **64** setzt simultan den Zustand einer Leitung FORMAT, um das Paritätsformat der Sendung auf der Leitung Rx anzugeben, das heißt, (mit Bezug auf Fig. 1), ob 8 Bits der Schriftzeichen gesendet sind, oder lediglich 7 Bits plus ein Paritätsbit P; der Zustand der Leitung FORMAT wird in dem Register **62** simultan mit den Auswahldaten BR gespeichert.

[0071] Fig. 7 ist ein Flussdiagramm der Operationen, die durch den Sequenzierer **10** der Schaltung von Fig. 6 ausgeführt werden, und erleichtert ein Verständnis der Operation der Schaltung.

[0072] **200**. Der Sequenzierer wartet auf das Setzen des Bits START; dieses Bit wird durch den Mikroprozessor gesetzt, wenn zum Beispiel ein Benutzer angibt, dass ein neues Terminal mit der Leitung Rx verbunden ist.

[0073] **202**. Das Bit START wird gesetzt. Der Sequenzierer wartet auf eine abfallende Flanke auf der Leitung Rx, entsprechend dem Auftreten eines Startbits für die Sendung eines Schriftzeichens (zum Beispiel bei dem Zeitpunkt t_0 von Fig. 2).

[0074] **204**. Ein Puls wird auf der Startleitung STARTC geliefert, wobei die Steuerschaltung **12** aufgefordert wird, eine Frequenzidentifizierungssequenz zu starten.

[0075] **206**. Der Sequenzierer wartet auf das Auftreten einer ansteigenden Flanke des Signals Rx.

[0076] **208**. Wenn die ansteigende Flanke auftritt, wartet der Sequenzierer auf die nächste abfallende Flanke des Signals Rx.

[0077] **210**. Wenn diese abfallende Flanke auftritt, liefert der Sequenzierer einen Puls STOPC zu der Steuerschaltung **12**, um die Frequenzidentifizierungssequenz zu stoppen.

[0078] In diesem Moment hat die Frequenzidentifizierungsschaltung die Schritte ausgeführt, die mit Bezug auf Fig. 4 beschrieben sind. Die Frequenz des Signals CK, das den UART steuert, ist, außer bei den Fehlerbedingungen, die durch das Signal REJ angegeben sind, die Frequenz, die der erwünschten Nenngeschwindigkeit zugeordnet ist.

[0079] **212**. Eine Überprüfung wird vorgenommen, ob das Unterdrückungssignal REJ gesetzt ist. Der Zweck besteht darin, zu wissen, ob die zweite abfallende Flanke innerhalb eines tolerierbaren Fehlerbereichs ordnungsgemäß auftrat, das heißt ob die vorliegende Frequenz des Taktsignals CK angemessen ist.

[0080] **214**. Das Signal REJ ist nicht gesetzt, was angibt, dass eine a priori gültige Sendegeschwindigkeit gefunden wurde. Das Signal GO zum Starten des UART ist zu einem aktiven Zustand (1) gesetzt. Der UART ist somit theoretisch eingerichtet und wartet auf ein Sendestartbit (START) vor einem Erfassen der acht folgenden Bits bei der Geschwindigkeit, die durch das Taktsignal desselben eingerichtet ist, in dem vorliegenden Fall durch das vorliegende Signal CK. Falls das Schriftzeichen, das zu der Leitung Rx gesendet wird, das gewöhnliche Schriftzeichen CR ist, das geliefert wird, um die Geschwindigkeit zu identifizieren (Fig. 1), identifiziert der UART das zweite Bit dieses Schriftzeichens als das Startbit und startet ein Erfassen der acht folgenden Bits 11000, P, STOP und 1.

[0081] **216**. Der Sequenzierer wartet auf das Setzen der Leitung CAVAIL, was angibt, dass acht Bits eines Schriftzeichens durch den UART erfasst wurden.

[0082] **218**. Die acht Bits des Schriftzeichens wurden durch den UART erfasst und werden bei dem Eingang der Prüfschaltung **64** bereitgestellt, die durch das Signal CAVAIL freigegeben ist. Die Prüfschaltung **64** vergleicht die acht Bits mit zwei möglichen erwünschten Werten, einer ohne ein Paritätsbit 11000011, der andere mit einem Paritätsbit 11000111. Falls die erfassten Bits einem dieser Werte entsprechen, wird das Signal OK gesetzt und die Leitung FORMAT wird zu einem Zustand gesetzt, zum Beispiel dem Zustand des Paritätsbits, was angibt, dass die Sendung mit oder ohne ein Paritätsbit ausgeführt wird.

[0083] **220**. Das Signal OK ist gesetzt. Der UART ist zu der geeigneten Sendegeschwindigkeit gesetzt. Die gefundenen Ergebnisse (die Geschwindigkeit, die durch die Daten BR geliefert wird, die Parität, die durch die Leitung FORMAT geliefert wird) werden in dem Register **62** durch das Setzen der Leitung STORE gespeichert. Ein Unterbrechungsbit IRQ wird freigegeben, um zu bewirken, dass der Mikroprozessor die gefundenen Ergebnisse speichert. Der Sequenzierer kehrt dann zu dem Wartemodus zurück und wartet auf ein neues Setzen des Bits START zu 1 bei dem Block **200**.

[0084] **222**. Das Schriftzeichen, das durch den UART erfasst wird, entspricht nicht dem erwünschten Schriftzeichen. Dies kann von einem falschen Identifizierungsschriftzeichen, das gesendet wurde, oder einer falschen Sendegeschwindigkeit, die identifiziert wurde, herrühren.

[0085] **224**. Das System wartet, bis die Leitung Rx zu dem Bereitschaftszustand derselben zurückkehrt, das heißt dem Zustand 1, und wartet auf die nächste Sendung eines Schriftzeichens zu dem Block **202**. Die gleiche Operation wird ausgeführt, falls bei dem Block **212** das Fehlersignal REJ gesetzt ist, das heißt falls die Sendefrequenz unbekannt ist.

[0086] Falls das Signal BR = 0 zwischen den Blöcken **204** und **210** gesetzt wird, was das Setzen des Signals REJ bewirkt, ist entweder die Sendegeschwindigkeit zu niedrig oder das Signal Rx befindet sich ständig in einem unerwarteten Zustand. Wenn ein angemessener Zustand des Signals Rx wieder eingerichtet ist, wird eine Ausführung durch ein Überprüfen des Signals REJ bei dem Block **212** und ein Warten auf die nächste ansteigende Flanke der Leitung Rx bei dem Block **224** fortgesetzt.

[0087] Bei einer selbstanpassenden Schaltung, wie beispielsweise derselben, die in Fig. 6 beschrieben ist, ist es möglich, Sendegeschwindigkeiten in Höhe von 921600 Bit/Sekunde oder sogar mehr zu erfassen, falls die Technologie der Schaltungen es möglich macht; im Gegensatz zu herkömmlichen Softwarelösungen ist es nicht möglich, Geschwindigkeiten von mehr als 19200 Bit/Sekunde zu erfassen, ohne Gefahr zu laufen, alle existierenden seriellen Verbindungen unangemessen zu steuern.

[0088] Der Sequenzierer **10** und die Steuerschaltung **12** können durch Fachleute auf dem Gebiet ohne weiteres mit der Hilfe der entsprechenden Flussdiagramme gefertigt werden. Genauer gesagt können die Flussdiagramme direkt in eine fortgeschrittene Logikschaltung-Beschreibungssprache umgeschrieben werden, wie beispielsweise die VHDL, die ein direktes Programmieren von Logikschaltungen wie beispielsweise PALs ermöglicht.

[0089] Wie es Fachleuten auf dem Gebiet ersichtlich ist, können verschiedene Modifikationen an den obigen offenbarten bevorzugten Ausführungsbeispielen vorgenommen werden, genauer gesagt an den Flussdiagrammen, die unterschiedlich eingerichtet sein können, während dieselben die gleichen Funktionen liefern.

[0090] Obwohl eine Anwendung der Erfindung auf eine Sendung auf seriellen Leitungen beschrieben wurde, gilt die Erfindung in jeglichen Fällen, wenn es notwendig ist, eine Frequenz unter mehreren möglichen diskreten Frequenzen zu identifizieren, wie auch immer die Variationen der diskreten Frequenzen mit Bezug aufeinander sein mögen.

[0091] In der vorhergehenden Beschreibung wird

angenommen, dass die Sendung der i + 1-ten Pulsfolge bei dem maximalen Zeitpunkt t_i (i = 0, 1, 2, ...) beginnt. In der Tat ist alles, was erforderlich ist, dass der Wechsel zwischen Taktpulsfolgen nicht während dem Intervall t_i' bis t_i auftritt, wenn das Messzeitintervall in Bezug auf ein Logiksignal bei der i-ten diskreten Frequenz gültig enden kann; falls diese Bedingung eingehalten wird, kann die Taktpulsfolgenfrequenz am Ende des Messintervalls als eine Angabe der Nennfrequenz des Logiksignals verwendet werden.

[0092] Somit könnte zum Beispiel die Sendung der i + 1-ten Pulsfolge zu dem Zeitpunkt t_i' begonnen werden; falls das Ende des Messintervalls während der Sendung der j-ten Pulsfolge (j = 1, 2, ...) auftritt, ist in diesem Fall die erwünschte Nennfrequenz die j – 1-te an Stelle der j-ten Frequenz (es ist ersichtlich, dass bei dieser Anordnung die erste Pulsfolge keinem anderen Zweck dient als das Intervall t_1' zeitlich zu bestimmen, was natürlich auf andere Weisen vorgenommen werden kann – es ist deshalb gleichermaßen gültig, die Pulsfolge, die bei t_1' beginnt, als die erste Pulsfolge zu betrachten, in welchem Fall die erwünschte Nennfrequenz die j-te Frequenz ist). In jedem Fall sollte eine endgültige Überprüfung vorgenommen werden, dass das Ende des Messintervalls in der aktuellen Pulsfolge vor dem Zeitpunkt t_j auftrat.

[0093] Bei der vorhergehenden Anordnung eines Beginnens einer neuen Pulsfolge bei dem minimalen Zeitpunkt t_i' ist es immer noch lediglich notwendig, zwei Zählwerte für jede Pulsfolge in dem Verifizierer **22** zu speichern, wobei dieselben Werte sind, die dem Zeitpunkt t_i und t_{i+1}' entsprechen. Falls der Wechsel zwischen Pulsfolgen zu einem gewissen Zeitpunkt zwischen dem Zeitpunkt t_i und t_{i+1}' vorgenommen wird, dann ist es jedoch notwendig, drei Zählwerte für jede Pulsfolge zu speichern, das heißt Zählwerte, die t_i, t_{i+1}' und dem erwünschten Wechselzeitpunkt (vor t_{i+1}') entsprechen.

[0094] Während es bevorzugt ist, dass das Verhältnis zwischen der Frequenz jeder Taktpulsfolge und der zugeordneten diskreten Nennfrequenz für alle Pulsfolgen das Gleiche ist, ist es ersichtlich, dass dies nicht notwendig ist, wenn jegliche Differenzen bei diesem Verhältnis einfach in den Zählwerten, die in dem Verifizierer **22** gespeichert sind, reflektiert sind.

Patentansprüche

1. Ein Verfahren zum Identifizieren der Nennfrequenz eines Logiksignals (Rx), das eine Frequenz aufweisen kann, die, mit einem möglichen Fehler, gleich einer von n diskreten Frequenzen mit sich fortlaufend verringerndem Wert von der ersten bis zu der n-ten Frequenz ist, wobei der mögliche Fehler, für jede der diskreten Frequenzen, einen vorbestimmten zulässigen Bereich aufweist, der durch einen maximalen positiven und einen maximalen negativen Fehlerwert definiert ist, wobei das Verfahren während einer Messungsperiode implementiert wird, die durch zwei vorbestimmte Flanken des Logiksignals (Rx) eingerichtet ist, **dadurch gekennzeichnet**, dass das Verfahren die folgenden Schritte umfasst:
(c) zu dem Ausmaß, das durch die Dauer der Messungsperiode gestattet ist, sequentielles Senden von Folgen von Taktpulsen, wobei jede aufeinanderfolgende Taktpulsfolge eine geringere Frequenz als die Vorgängerin derselben aufweist und eine jeweilige vorbestimmte Zeitdauer aufweist, wobei die vorbestimmten Zeitdauern der Taktpulsfolgen derart sind, dass:
(i) die Gesamtdauer der Pulsfolgen von der ersten zu der i-ten Pulsfolge, für 0 < i < n, zumindest so groß wie die Dauer der Messungsperiode ist, die in dem Fall resultieren würde, dass das Logiksignal die i-ten diskreten Frequenzen aufweist, die durch den entsprechenden maximalen negativen Fehlerwert verringert sind, aber geringer als die Dauer der Messungsperiode ist, die in dem Fall resultieren würde, dass das Logiksignal die (i + i)-te diskrete Frequenz aufweist, die durch den entsprechenden maximalen positiven Fehlerwert erhöht ist,
(ii) die Gesamtdauer der Pulsfolgen von der ersten zu der n-ten Pulsfolge zumindest so groß wie die Dauer der Messungsperiode ist, die in dem Fall resultieren würde, dass das Logiksignal die n-te diskrete Frequenz aufweist, die durch den entsprechenden maximalen positiven Fehlerwert verringert ist,
(d) bei dem Auftreten der zweiten der vorbestimmten Flanken, Nehmen einer Angabe der dann aktuellen Taktfrequenz als eine Angabe der Nennfrequenz des Logiksignals.

2. Ein Verfahren gemäß Anspruch 1, bei dem die Zeitdauern der Taktpulsfolgen derart sind, dass die Summe der Zeitdauern der ersten bis zu der i-ten Folge von Taktpulsfolgen, für 0 < i ≤ n, im Wesentlichen gleich der Dauer der Messungsperiode ist, die in dem Fall resultieren würde, dass das Logiksignal eine Frequenz aufweist, die gleich der i-ten diskreten Frequenz ist, die durch den entsprechenden maximalen negativen Fehlerwert verringert ist.

3. Ein Verfahren gemäß Anspruch 2, das ferner den Schritt eines Angebens umfasst, dass die Nennfrequenz des Logiksignals unbekannt ist, falls die zweite der Flanken vor dem Ende der Sendung der n-ten Taktpulsfolge nicht auftritt.

4. Ein Verfahren gemäß Anspruch 2, das ferner den Schritt eines Angebens umfasst, dass die Nennfrequenz des Logiksignals unbekannt ist, falls, wenn die zweite der Flanken während der i-ten Pulsfolge für 0 < i ≤ n auftritt, die Dauer der tatsächlichen Messungsperiode geringer als die Dauer der Messungsperiode ist, die in dem Fall resultieren würde, dass

das Logiksignal eine Frequenz aufweist, die gleich der i-ten diskreten Frequenz ist, die durch den entsprechenden maximalen positiven Fehlerwert erhöht ist.

5. Ein Verfahren gemäß Anspruch 1, bei dem das Verhältnis des Werts der i-ten diskreten Frequenz zu dem Frequenzwert der i-ten Taktpulsfolge für alle Werte von i für $0 < i \leq n$ das gleiche ist.

6. Eine Frequenzidentifizierungsschaltung zum Identifizieren der augenblicklichen Nennfrequenz eines Logiksignals (Rx), das eine augenblickliche Frequenz aufweisen kann, die mit einem möglichen Fehler, gleich einer von einer Mehrzahl von diskreten Nennfrequenzen ist, dadurch gekennzeichnet, dass die Schaltung folgende Merkmale aufweist:
– einen Taktgenerator (**18**) zum Liefern einer Mehrzahl von Taktsignalen, wobei jedes Taktsignal eine Frequenz aufweist, die ein konstantes Mehrfaches einer zugeordneten der diskreten Frequenz ist;
– einen Sequenzierer (**10**) zum Liefern eines Startsignals (STARTC) bei dem Auftreten einer ersten vorbestimmten Flanke des Logiksignals und eines Stoppsignals (STOPC) bei dem Auftreten einer zweiten vorbestimmten Flanke des Logiksignals;
– einen ersten Zähler (**14**) zum Enthalten von Auswahldaten (BR), deren aufeinanderfolgende Werte jeweiligen Taktsignalen von sich fortlaufend verringernden Frequenzen und somit der jeweiligen diskreten Frequenz mit sich diskreten Frequenz mit sich fortlaufend verringerndem Wert zugeordnet sind;
– einen Multiplexer (**16**), der mit dem Taktgenerator (**18**) verbunden ist, und durch die Auswahldaten gesteuert ist, um das zugeordnete Taktsignal (CK) zu liefern;
– einen zweiten Zähler (**20**) zum Empfangen des Ausgangs (CK) des Multiplexers an einem Zähleingang;
– einen Komparator (**22**) zum Liefern eines Freigabesignals (L = MAX), wenn der Inhalt (L) des zweiten Zählers einen Maximalwert erreicht, der in Abhängigkeit von den Auswahldaten (BR) ausgewählt ist; und
– eine Steuerschaltung (**12**) zum Liefern von Initialisierungssignalen (INITBR, RSTL) zu dem ersten und dem zweiten Zähler, sobald dieselbe das Startsignal (STARTC) empfängt, und zum Liefern eines Zählfreigabesignals (DECBR) zu dem ersten Zähler und eines Initialisierungssignals (RSTL) zu dem zweiten Zähler, wenn das Freigabesignal (L = MAX) durch den Komparator geliefert wird, wobei die Steuerschaltung in einen Bereitschaftszustand geht, wenn dieselbe das Stopsignal (STOPC) empfängt.

7. Eine Frequenzidentifizierungsschaltung gemäß Anspruch 6, bei der der Maximalwert, der in Abhängigkeit von den Auswahldaten (BR) ausgewählt ist, dem Wert entspricht, der durch den zweiten Zähler (**20**) bei dem Auftreten der zweiten vorbestimmten Flanke des Logiksignals in dem Fall erreicht worden wäre, dass letzteres die diskrete Frequenz aufweist, die den vorliegenden Auswahldaten (BR) zugeordnet ist, verringert durch einen maximalen zulässigen Fehler.

8. Eine Frequenzidentifizierungsschaltung gemäß Anspruch 7, bei der der Komparator (**22**) ein Unidentifizierte-Frequenz-Signal (REJ) liefert, das angibt, dass die Frequenz des Logiksignals nicht einer diskreten Nennfrequenz entspricht, wenn eines der folgenden auftritt:
– die Auswahldaten (BR) erreichen einen Wert jenseits dem, der der niedrigsten diskreten Nennfrequenz zugeordnet ist,
– der Inhalt (L) des zweiten Zählers ist niedriger als ein Minimalwert, der dem Wert entspricht, der durch den zweiten Zähler bei dem Auftreten der zweiten vorbestimmten Flanke des Logiksignals in dem Fall erreicht worden wäre, dass letzteres die diskrete Frequenz aufweist, die den vorliegenden Auswahldaten (BR) zugeordnet ist, erhöht durch einen maximalen zulässigen Fehler.

9. Eine Frequenzidentifizierungsschaltung gemäß Anspruch 8, bei der der Maximal- und der Minimalwert, die den Werten der Auswahldaten (BR) entsprechen, in jeweiligen Speichern (**30**, **31**) gespeichert sind, die durch die Auswahldaten adressiert werden, wobei der Wert, der durch jeden Speicher geliefert wird, mit dem Ausgang (L) des zweiten Zählers (**20**) in einem jeweiligen Komparator (**34**, **35**) verglichen wird.

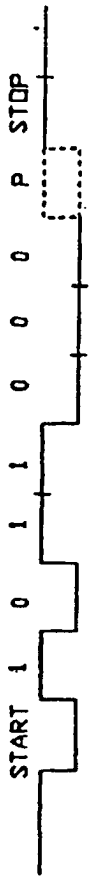
10. Eine Schaltung, die sich selbst an die Sendegeschwindigkeit einer Vorrichtung anpasst, die mit einer seriellen Leitung (Rx) verbunden ist, auf der die Vorrichtung ein vorbestimmtes Schriftzeichen für die Erfassung der Sendegeschwindigkeit sendet, die folgende Merkmale umfasst:
– die Frequenzidentifizierungsschaltung gemäß Anspruch 8;
– einen universellen asynchronen Empfänger-Sender (UART = Universal Asynchronous Receiver Transmitter) (**60**), der mit der seriellen Leitung (Rx) verbunden ist und durch das Taktsignal (CK) getaktet ist, das durch den Multiplexer der Identifizierungsschaltung geliefert wird; und
– eine Überprüfungsschaltung (**64**) zum Liefern eines Korrektes-Schriftzeichen-Signals (OK), falls das Schriftzeichen, das durch den UART empfangen wird, gleich dem vorbestimmten Schriftzeichen ist;
wobei der Sequenzierer (**10**) der Frequenzidentifizierungsschaltung mit der seriellen Leitung (Rx) zum Liefern des Startsignals (STARTC) bei dem Auftreten einer ersten Flanke an der seriellen Leitung, die dem ersten Bit des vorbestimmten Schriftzeichen entspricht, und zum Liefern des Stoppsignals (STOPC) bei dem Auftreten einer zweiten Flanke an der seriellen Leitung, die einem vorbestimmten nachfolgenden Bit des vorbestimmten Schriftzeichens entspricht,

verbunden ist, wobei der Sequenzierer (**10**) ferner wirksam ist, um ein weiteres Startsignal (GO) zu dem UART zu liefern, wenn das Unidentifizierte-Frequenz-Signal (REJ) nach dem Auftreten des Stop-Signals (STOPC) nicht vorhanden ist, und um das Taktsignal, das zu dem UART geliefert wird, unverändert beizubehalten, wenn das Korrektes-Schriftzeichen-Signal (OK) durch die Überprüfungsschaltung am Ende der Sendung des vorbestimmten Schriftzeichens geliefert wird.

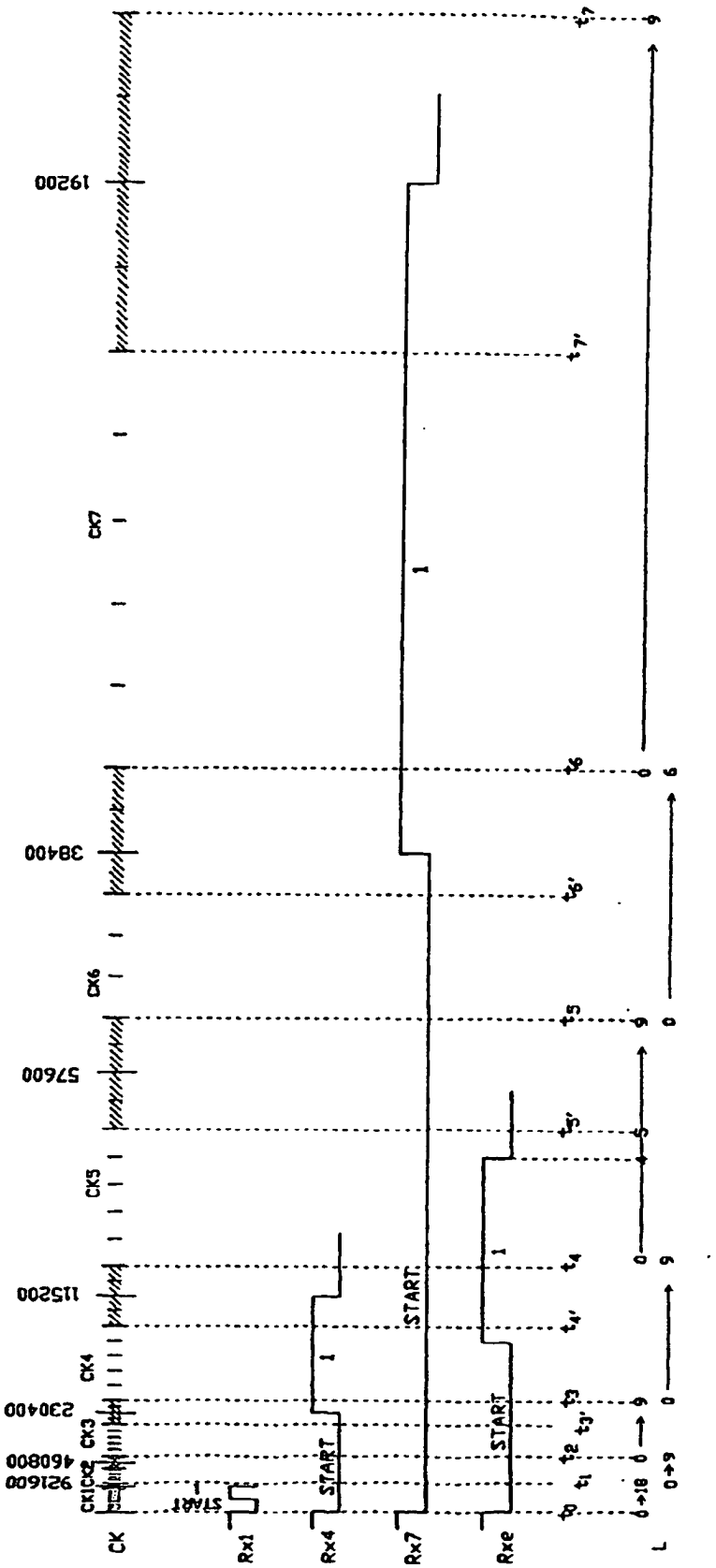
Es folgen 5 Blatt Zeichnungen

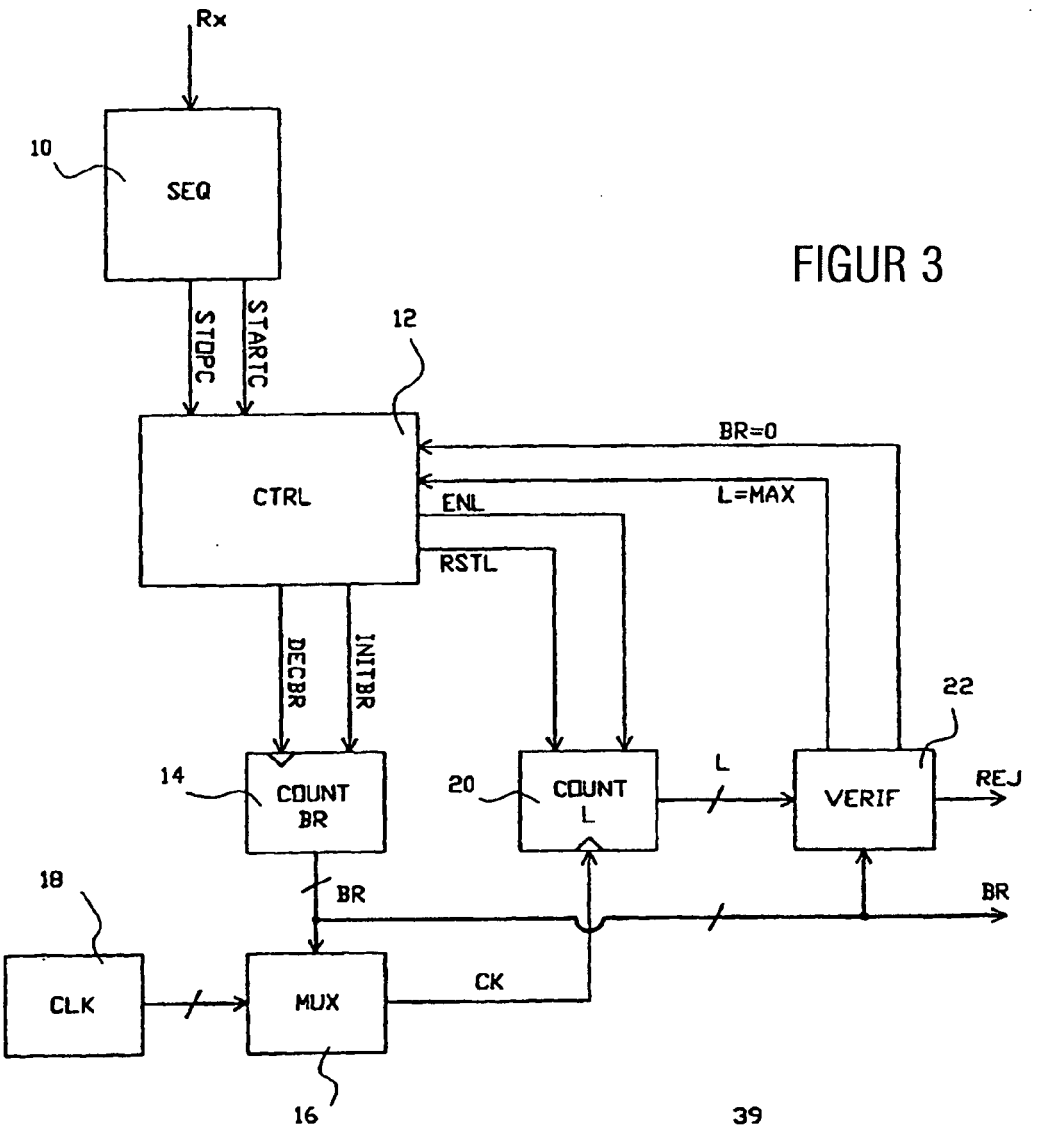
Anhängende Zeichnungen

FIGUR 1

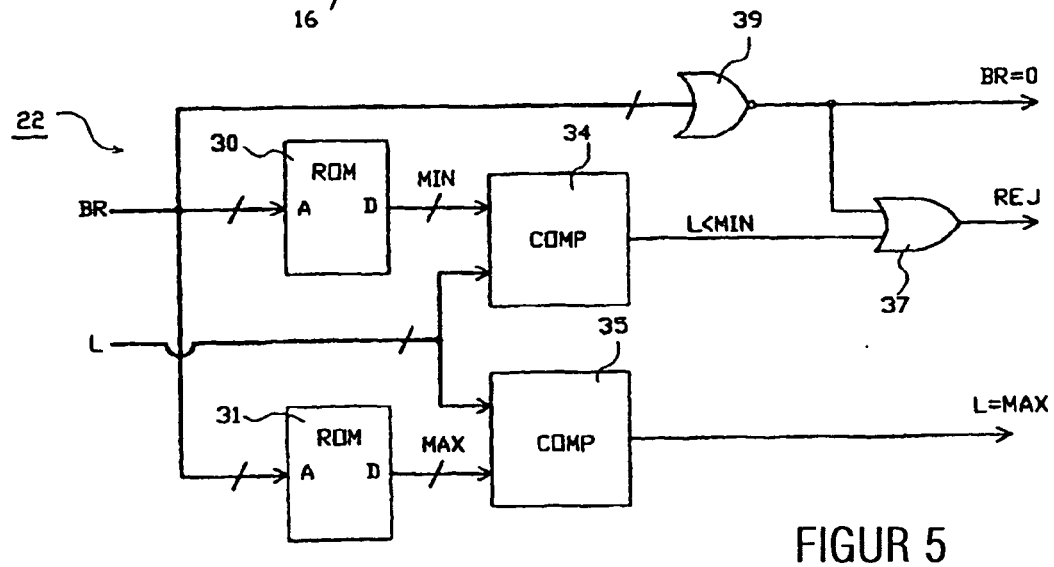


FIGUR 2

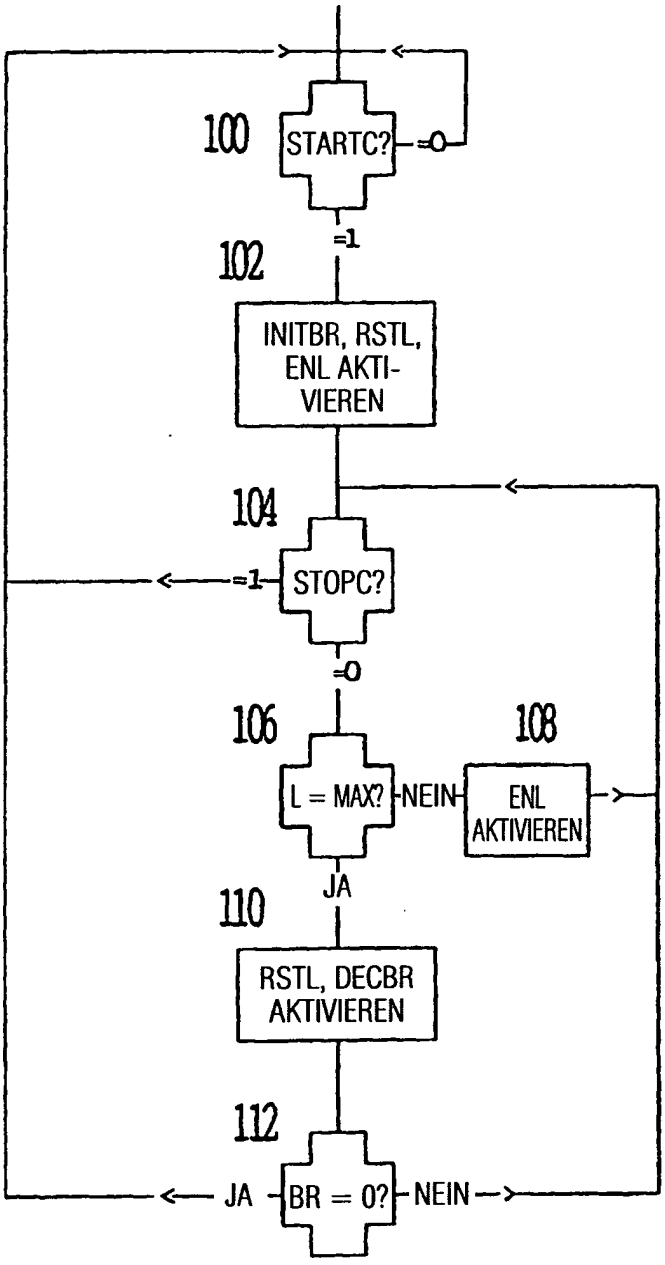




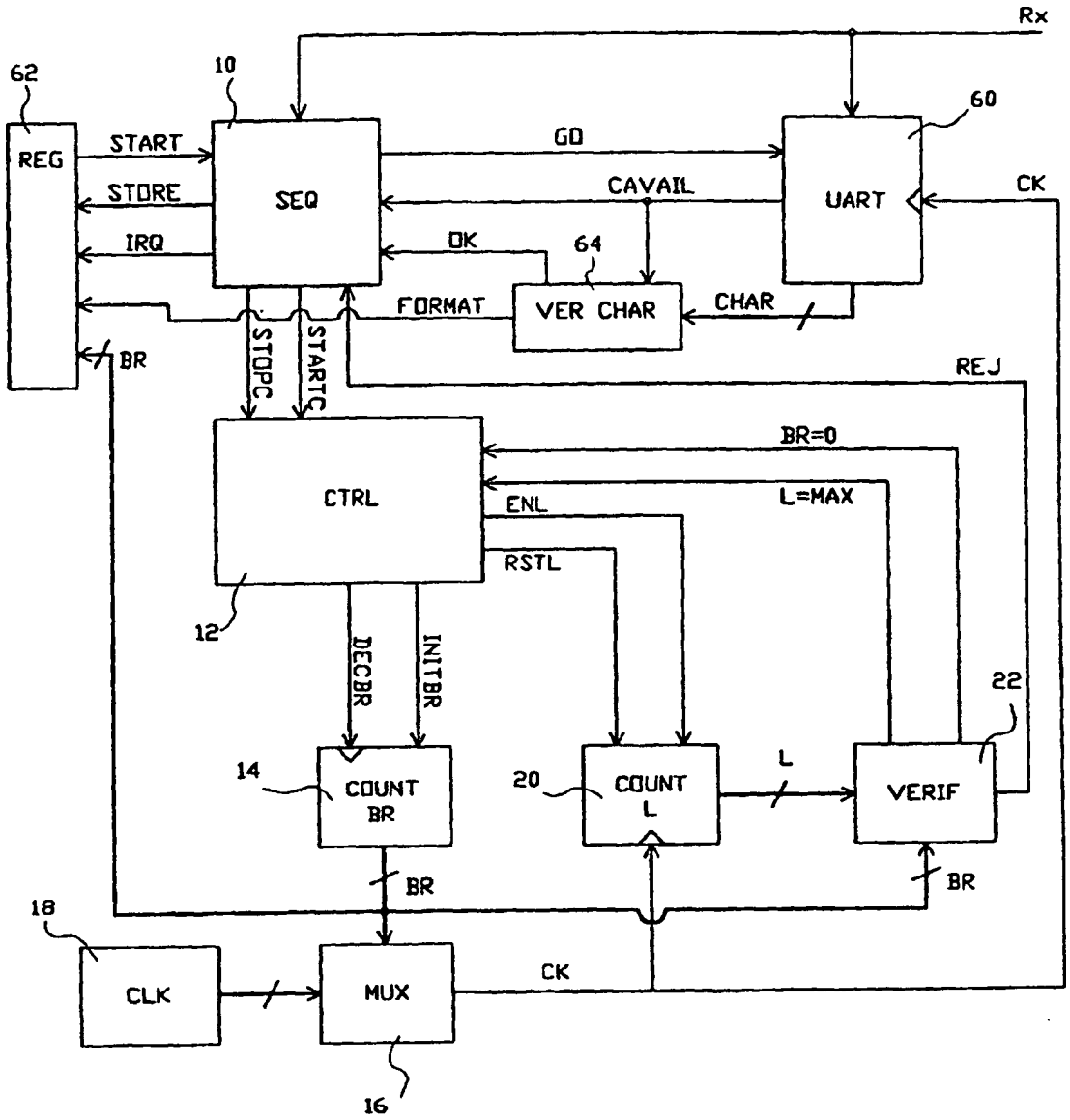
FIGUR 3



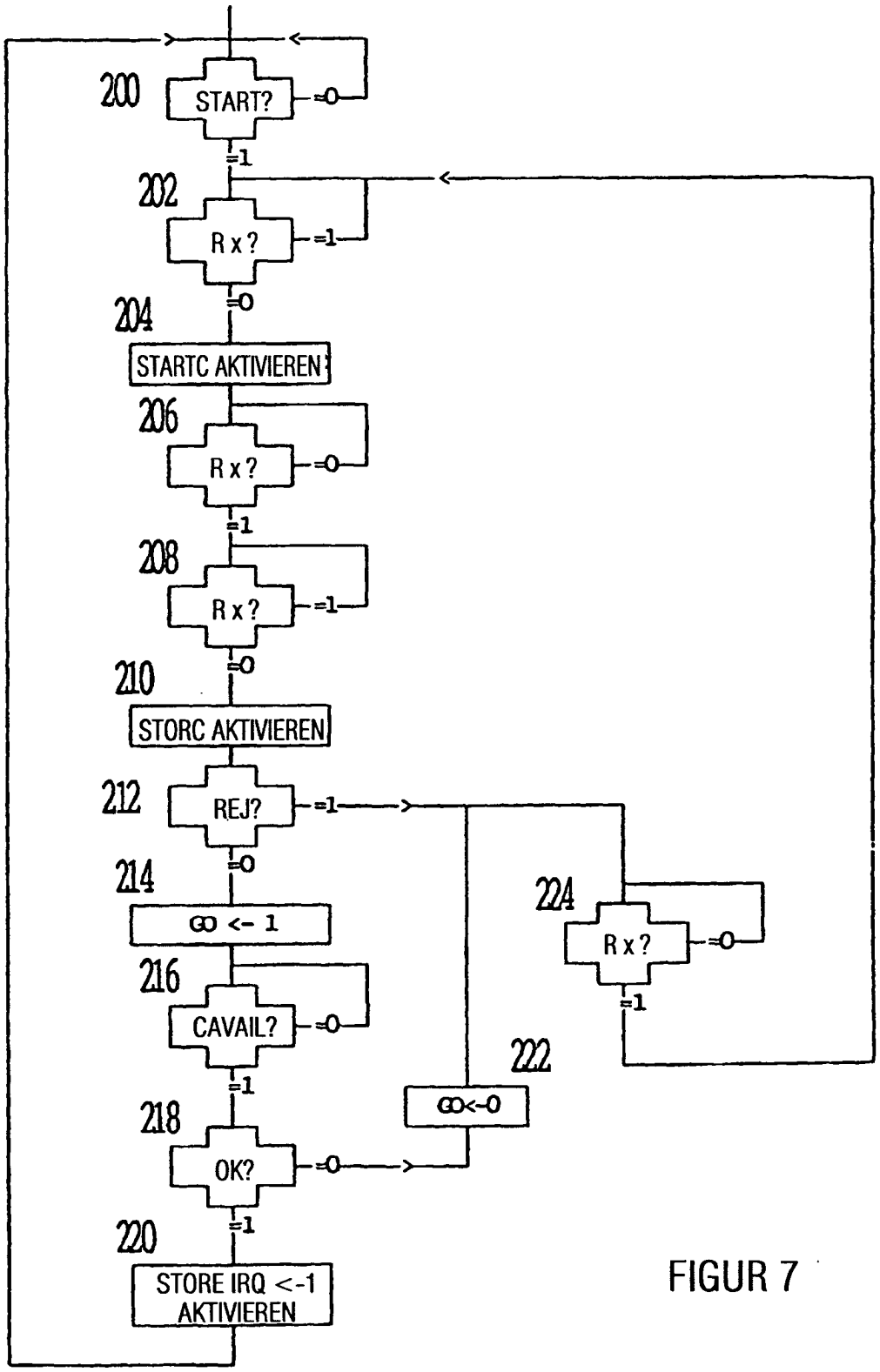
FIGUR 5


100
STARTC?
=0
=1
102
INITBR, RSTL, ENL AKTI-VIEREN
104
=1
STOPC?
=0
106
L = MAX?
NEIN
108
ENL AKTIVIEREN
JA
110
RSTL, DECBR AKTIVIEREN
112
JA
BR = 0?
NEIN

FIGUR 4


Rx
62
10
60
REG
START
SEQ
GO
UART
CK
STORE
CAVAIL
IRQ
OK
64
FORMAT
VER CHAR
CHAR
BR
STOPC
STARTC
REJ
BR=0
CTRL
L=MAX
ENL
RSTL
12
DECBR
INITBR
22
14
COUNT BR
20
COUNT L
L
VERIF
BR
BR
18
CLK
MUX
CK
16

FIGUR 6


200
START?
=0
=1
202
R x ?
=1
=0
204
STARTC AKTIVIEREN
206
R x ?
=0
=1
208
R x ?
=1
=0
210
STORC AKTIVIEREN
212
REJ?
=1
=0
214
GO <- 1
216
CAVAIL?
=0
=1
218
OK?
=0
=1
220
STORE IRQ <-1
AKTIVIEREN
222
GO<-0
224
R x ?
=0
=1

FIGUR 7