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

(10) **DE 600 22 941 T2** 2006.06.29

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 1 054 525 B1**
(21) Deutsches Aktenzeichen: **600 22 941.6**
(96) Europäisches Aktenzeichen: **00 303 894.0**
(96) Europäischer Anmeldetag: **09.05.2000**
(97) Erstveröffentlichung durch das EPA: **22.11.2000**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **05.10.2005**
(47) Veröffentlichungstag im Patentblatt: **29.06.2006**

(51) Int Cl.[8]: ***H04L 1/00*** (2006.01)
H04L 7/00 (2006.01)
H04J 3/06 (2006.01)
H04L 7/04 (2006.01)

(30) Unionspriorität:
312711 **17.05.1999** **US**

(84) Benannte Vertragsstaaten:
DE, FR, GB

(73) Patentinhaber:
Lucent Technologies Inc., Murray Hill, N.J., US

(74) Vertreter:
derzeit kein Vertreter bestellt

(72) Erfinder:
Anderlind, Erik Einar, Parsippany, New Jersey 07054, US; Mailaender, Laurence Eugene, New York, US

(54) Bezeichnung: **Verfahren und Verfahren zur Übertragung von codierten Daten mit variabler Länge in einer Umgebung mit geringem Störabstand**

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 betrifft allgemein die Übertragung von Daten in einer Umgebung mit geringem Störabstand und insbesondere die drahtlose Übertragung von Daten.

BESCHREIBUNG DES STANDES DER TECHNIK

[0002] Die Übertragung von Daten in Umgebungen oder Medien mit geringem Störabstand erfolgt derzeit auf der Basis einer festen Rahmenlänge oder einer festen Rahmendauer. Eine feste Rahmendauer eignet sich zum Transportieren von Sprachdaten, da Sprachcodierer und -decodierer so eingestellt werden können, dass sie Datenabtastwerte mit fester Länge (oder eine kleine Gruppe von Abtastwertlängen) innerhalb des Rahmenintervalls erzeugen. Aber für den Datenverkehr gibt es ein breites Spektrum möglicher Paketlängen, und eine Nichtübereinstimmung zwischen der Paketlänge und einer Rahmenlänge muss auf die eine oder andere Art kompensiert werden. In der Regel werden Paketsegmentierung und Rahmenauffüllung verwendet, um Daten in eine feste Rahmengröße hineinzupressen. Dementsprechend wäre es wünschenswert, wenn man Datenrahmen mit variabler Länge versenden könnte.

[0003] Eine Umgebung, in der zur Zeit Informationspakete mit variabler Länge verwendet werden, ist die Computernetzwerkumgebung. Wie in der den Stand der Technik darstellenden Fig. 1 gezeigt ist, enthält eine typische Netzwerkpaketstruktur eine Präambel **3** (zur Synchronisation), eine Zieladresse **5**, einen Paketgrößenindikator **7**, die Daten **9** und ein Fehlerprüffeld **11**. Bei verdrahteten Computernetzwerkanwendungen wird in der Regel davon ausgegangen, dass der Empfänger den Anfang und das Ende jedes Paketes problemlos erkennen kann. Das liegt daran, dass der Störabstand auf dem drahtgebundenen Übertragungskanal hoch ist.

[0004] Eine kurze Präambel **3** lässt dem Empfänger Zeit, den Anfang eines Paketes zu finden, Zeitsynchronisation herzustellen und die Paketgröße zu lesen. Der Paketgrößenindikator **7** zeigt die Größe des Datenfeldes **9** an und gestattet es dem System praktisch, einen Endpunkt des Paketes im Voraus zu bestimmen. Das Fehlerprüffeld **11** dient dazu, Informationen in dem Paket, einschließlich des Paketgrößenindikators **7**, zu schützen. Wenn das Fehlerprüffeld **11** versagt, so wird das Paket verworfen.

[0005] In drahtlosen Netzwerken und anderen Umgebungen mit geringem Störabstand können sich Paketköpfe mit einem Format, das beispielsweise den Größenindikator **7** und das Fehlerprüffeld **11** enthält, als unzuverlässig erweisen (d. h. Fehler enthalten), und viele Informationspakete müssten verworfen werden. Dementsprechend ist eine Paketstruktur, wie sie in der den Stand der Technik darstellenden Fig. 1 gezeigt ist, für ein drahtloses Netz und andere Umgebungen mit geringem Störabstand ungeeignet.

[0006] Insbesondere erfordern Umgebungen mit geringem Störabstand den Einsatz hoch-entwickelter Fehlerkorrekturcodes. Bei einem sehr geringen Störabstand werden in der Regel codierte Symbole empfangen, wodurch das Erkennen des Anfangs und des Endes des Paketes erschwert wird. Des Weiteren sind die Paketgrößeninformationen in dem Paketkopf codiert und können erst gelesen werden, nachdem die Paketgrenzen ermittelt wurden. Ohne eine zuverlässige Erkennung der Paketgrenzen kann die Datenkommunikation nicht weitergehen. Aus diesem Grund wurde bisher noch keine Übertragung von Paketen mit variabler Länge in Umgebungen mit geringem Störabstand realisiert.

[0007] In solchen drahtlosen Netzwerksystemen werden die Informationen in der Regel redundant codiert, damit, wenn Fehlerbits eingetragen werden, die Informationen fehlerfrei rekonstruiert werden können. Bei typischen Anwendungen mit Mehrfachzugriff im Codemultiplex (CDMA) enthält die codierte Nachricht, wenn die ursprüngliche Nachricht "N" Bits enthält, "2N" oder "3N" Bits.

[0008] A. S. Tannenbaum: Computer Networks, Dritte Ausgabe, 1996, Prentice-Hall International XP002242058, betrifft Computernetzwerke. Es offenbart eine Bitstopftechnik, wobei jedes Mal, wenn die Sicherungsschicht eines Absenders auf fünf aufeinanderfolgende "1" in Daten trifft, ein System automatisch ein "0"-Bit in den ausgehenden Bitstrom stopft. Dieses Bitstopfen ist analog dem Zeichenstopfen, wobei eine Datenübertragungsumschaltung (DLE) in einen ausgehenden Zeichenstrom vor der DLE in die Daten gestopft wird. Wenn der Empfänger fünf aufeinanderfolgende eingehende "1"-Bits sieht, auf die ein "0"-Bit folgt, so löscht das System automatisch das "0"-Bit. Beim Bitstopfen kann die Grenze zwischen zwei Rahmen durch ein Markierungsmuster erkannt werden.

KURZDARSTELLUNG DER ERFINDUNG

[0009] Das Verfahren und die Vorrichtung gemäß der Erfindung sind so, wie in den angehängten Ansprüchen dargelegt. Bevorzugte Ausführungsformen sind in den abhängigen Ansprüchen dargelegt.

[0010] Unter Verwendung der Vorrichtung und des Verfahrens der vorliegenden Erfindung können codierte Daten mit variabler Länge in Umgebungen mit geringem Störabstand übertragen werden. Durch die Vorrichtung und das Verfahren der vorliegenden Anmeldung wird an einen Anfangsteil von codierten Da-

ten ein bestimmtes Datenmuster hinzugefügt, um einen Anfang der codierten Daten anzuzeigen, und das gleiche oder ein anderes Datenmuster wird dem Endteil der codierten Daten hinzugefügt, um das Ende der codierten Daten anzuzeigen. Weil die codierten Daten selbst das Anfangs- oder End-Datenmuster oder Varianten davon enthalten können und darum irrtümlicherweise einen Anfang oder ein Ende der codierten Daten anzeigen können, werden des Weiteren die codierten Daten zuerst auf solche Muster überprüft. Wenn ein solches Muster oder ähnliche Muster, die in Umgebungen mit geringem Störabstand schnell einmal falsch interpretiert werden können, in den codierten Daten festgestellt werden, so werden diese Muster durch neue Muster substituiert. Dadurch können codierte Daten mit variabler Länge in einer Umgebung mit geringem Störabstand übertragen werden und können anschließend problemlos decodiert werden.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

[0011] Die Merkmale, Aspekte und Vorteile der vorliegenden Erfindung werden anhand der folgenden Beschreibung, der angehängten Ansprüche und der begleitenden Zeichnungen, in denen gleiche Zahlen gleiche Elemente darstellen, besser verstanden.

[0012] Fig. 1 veranschaulicht ein zum Stand der Technik gehörendes Datenpaket zur Übertragung in einer Kommunikationsumgebung mit großem Störabstand.

[0013] Fig. 2A–Fig. 2C veranschaulichen Musterdatenpakete der vorliegenden Erfindung.

[0014] Fig. 3 veranschaulicht eine Datenmustersubstituierung der vorliegenden Erfindung.

[0015] Fig. 4 veranschaulicht eine Vorrichtung zum Erzeugen von codierten Daten mit variabler Länge zur Übertragung.

[0016] Fig. 5 veranschaulicht eine Vorrichtung zum Decodieren von übertragenen codierten Daten mit variabler Länge.

[0017] Fig. 6 veranschaulicht ein Verfahren – in einer bevorzugten Ausführungsform der vorliegenden Erfindung – zur Übertragung von codierten Daten mit variabler Länge.

[0018] Fig. 7 veranschaulicht – in einer bevorzugten Ausführungsform der vorliegenden Erfindung – ein Verfahren zum Decodieren von empfangenen codierten Daten mit variabler Länge.

[0019] Fig. 8 veranschaulicht mögliche Probleme bei der Datenmustersubstituierung.

[0020] Fig. 9 veranschaulicht eine Datenmustersubstituierung einer weiteren bevorzugten Ausführungsform der vorliegenden Erfindung.

DETAILLIERTE BESCHREIBUNG DER VORLIEGENDEN ERFINDUNG

[0021] Die vorliegende Erfindung verwendet ein bestimmtes Datenmuster und fügt es an den Anfang eines Datenpaketes an, um den Anfang von codierten Daten anzuzeigen, wenn diese in einer Umgebung mit geringem Störabstand übertragen werden. Das gleiche oder ein anderes Datenmuster wird einem Endteil der codierten Daten hinzugefügt, um das Ende von codierten Daten anzuzeigen, wenn sie in der Umgebung mit geringem Störabstand übertragen werden. Um zu gewährleisten, dass die codierten Daten selbst nicht das Anfangs- oder End-Datenmuster enthalten und somit fälschlicherweise den Anfang oder das Ende von codierten Daten anzeigen, werden die codierten Daten schließlich zuerst auf das Vorhandensein der Datenmuster oder ähnlicher Datenmuster geprüft, die als ein Anfangs- oder End-Indikator der codierten Daten verwendet werden. Wenn diese oder ähnliche Muster festgestellt werden, so werden sie durch andere Muster substituiert, um so ein falsches Erkennen eines Anfangs oder Endes der codierten Daten zu vermeiden.

[0022] Durch Verwenden eines bekannten oder vorgegebenen Datenmusters zum Anzeigen des Anfangs und des Endes von codierten Daten können Daten mit variabler Länge in solchen Datenpaketen übertragen werden. Des Weiteren wird, wenn diese Datenmuster oder die nächsten Nachbarn dieser Datenmuster, wenn sie auf natürliche Weise in den codierten Daten vorkommen, durch andere Datenmuster substituiert werden, eine Anzeige falscher Anfangs- und Endpunkte der übertragenen Daten vermieden. Dementsprechend ist es durch ein solches Regime möglich, codierte Daten mit variabler Länge in Umgebungen mit geringem Störabstand, wie beispielsweise in drahtlosen Netzen, zu übertragen.

[0023] Fig. 2A veranschaulicht ein typisches Datenpaket **20** der vorliegenden Erfindung. Das Datenpaket **20** enthält ein erstes Präambel-Datenmuster **22**, das beispielsweise mit "X" bezeichnet ist; ein Synchronisationsmuster **23**, das beispielsweise mit "S" bezeichnet ist; Daten **24** von einiger variierender Länge; und ein Postambel-Datenmuster **26**, das mit der Variable "Z" bezeichnet ist und an das Ende des Datenpaketes angefügt ist. Es ist zu beachten, dass "X" und "Z" generisch ausgewählt wurden, um ein erstes Präambel-Datenmuster bzw. ein zweites Postambel-Datenmuster zu bezeichnen, die den Anfangs- bzw. den Endteil von codierten Daten kenntlich machen, und in keiner Weise als Einschränkung anzusehen sind. Gleichermaßen stellt "S" generisch ein Synchronisations-Datenmuster dar, das nach dem

Präambel-Datenmuster "X" erkannt wird, wobei das Ende des Datenmusters "S" den Anfang einer codierten Datensequenz bezeichnet.

[0024] Bei einer bevorzugten Ausführungsform kann das Präambel-Datenmuster "X" in seiner einfachsten Form so gewählt werden, dass es mit dem Postambel-Datenmuster "Z" identisch ist. Besonders bevorzugt jedoch ist das Postambel-Datenmuster "Z" ein Teilmuster des Präambel-Datenmusters "X". Dadurch kann bei einer kontinuierlichen Übertragung von Daten, wie in Fig. 2C gezeigt, selbst wenn es zu Zeitsteuerungsfehlern kommt und das Datenmuster "X" nicht erkannt werden kann, vorzugsweise immer noch sein Teilmuster "Z" erkannt werden. Somit wird ein Ende eines Paketes erkannt, das in einer kontinuierlichen Datensequenz anzeigt, dass nach einem Datenmuster "S" und im Anschluss daran nach codierten Daten zu suchen ist.

[0025] Um die typischen Probleme von Umgebungen mit geringem Störabstand zu überwinden, wie beispielsweise Mehrfachnutzerinterferenzen oder geringe Signalleistung, wie man sie in der drahtlosen Kommunikation findet, beträgt das erste Datenmuster **22** ("X"), das dem Anfang des Datenpaketes **20** hinzugefügt wird, vorzugsweise etwa 8 oder mehr Symbole oder Bits. Durch Versuche wurde herausgefunden, dass 16 Bits (für die Datenmuster "S" und "X") eine zuverlässige Erkennung und Synchronisation bei einer Kommunikation über drahtlose Kanäle erbringen. Je nach Situation kann aber auch jede andere Anzahl von Bits oder Symbolen verwendet werden, solange das gewählte Datenmuster sowohl den entsprechenden Sender/Codierern als auch Empfänger/Decodierern bekannt ist.

[0026] Dementsprechend wird ein vorgegebenes Muster "X" von einer vorgegebenen Länge 1_X gewählt und in die Bitübertragungsschicht am Anfang jedes Datenpaketes **20** eingesetzt, wobei im Anschluss daran ein Synchronisationsmuster "S" eingesetzt wird. Das Datenmuster "Z" wird vorzugsweise dafür benutzt, das Ende des Datenpaketes **26** zu markieren, wie in Fig. 2A gezeigt, wobei das Datenmuster "Z" vorzugsweise wenigstens eine Teilmenge oder ein Teilmuster des Datenmusters "X" ist (d. h. X = 01000000 und Z = 0000). Des Weiteren wird das Datenmuster "Z" in der Regel so gewählt, dass es – trotz gelegentlich empfangener Symbolfehler und Bitverschiebungen in der Zeitsteuerung der Sequenz – leicht erkannt werden kann. Ein Beispiel für ein solches Datenmuster "Z" ist eines, das alle Einsen oder alle Nullen enthält.

[0027] Das Synchronisations-Datenmuster "S" folgt in der Regel dem Präambel-Datenmuster "X" und kann als ein zweiter Teil des ersten Datenmusters angesehen werden, der den Anfang von codierten Daten anzeigt. Das Datenmuster "S" wird so gewählt, dass es eine gute Synchronisation für den Anfang von codierten Daten ermöglicht. Beispiele für solche Datenmuster "S" sind die bekannten Barker-Sequenzen. Das Synchronisationsmuster "S" wird in der Regel als ein Datenmuster gewählt, das trotz gelegentlich empfangener Symbolfehler leicht erkennbar ist. Das Präambel-Datenmuster "X" gibt einem Empfänger zu erkennen, dass er nach dem Datenmuster "S" suchen soll, und das Datenmuster "S" synchronisiert den Empfänger so, dass, sobald das Datenmuster "S" empfangen wurde, codierte Daten beginnen. Natürlich ist anzumerken, dass auch noch andere Schichtebeneninformationen, einschließlich Adressinformationen oder Felder, zwischen das Datenmuster "S" und den Anfang der Daten eingefügt werden können (wobei zu beachten ist, dass diese sowohl dem Sender als auch dem Empfänger bekannt sein müssen).

[0028] Bei einer Ausführungsform der Erfindung wird das Datenmuster "X" so gewählt, dass es eine hohe Autokorrelation (geringe Auflösung) aufweist, wie beispielsweise eine nur aus Einsen bestehende Sequenz. Das Datenmuster "S" wird so gewählt, dass es eine geringe Autokorrelation (hohe Auflösung) aufweist, wie beispielsweise die oben erwähnten Barker-Sequenzen oder Zufallssequenzen.

[0029] Fig. 2B veranschaulicht ein weiteres Datenpaket **30**, das eine gewisse Zeit nach dem Datenpaket **20** gesendet wurde. Ähnlich dem Datenpaket **20** enthält auch das Datenpaket **30** das Datenmuster "X" am Anfang **32** der codierten Daten; ein Synchronisationsmuster "S"; einen Datenteil **34**, der kürzer oder von einer anderen Länge ist als der Datenteil **24** des Datenpaketes **20**; und das Datenmuster "Z", das dem Endteil **36** des Datenpaketes **30** hinzugefügt wurde. Wie in den Fig. 2A und Fig. 2B gezeigt, enthalten alle diskontinuierlich gesendeten Pakete, die in einer Luftschnittstelle mit "toter Luft" dazwischen übertragen wurden, separate Markierungen "X" bzw. "Z", die den Anfang und das Ende jedes Paketes bezeichnen.

[0030] Fig. 2C veranschaulicht eine Situation mit kontinuierlich gesendeten Datenpaketen **40**. Das Datenmuster "X" ist am Anfang **42** des ersten Datenpaketes, das Daten **46a** enthält, enthalten, wobei das Datenmuster "S" anschließend bei **43** eingefügt ist, was den Anfang der Datensequenz anzeigt. Ein weiteres Datenmuster "Z" ist am Endteil **44** des letzten der kontinuierlich gesendeten Datenpakete, die Daten **46c** enthalten, eingefügt. Bei einer solchen kontinuierlichen Übertragung von Daten dienen das Datenmuster "X" und das Datenmuster "S", die mit **48a, 49a** und **48b**, **49b** bezeichnet sind, dazu, sowohl das Ende eines Teils einer Gruppe von Daten als auch den Anfang einer anderen Gruppe anzuzeigen. Beispielsweise zeigen die mit **48a**, **49a** bezeichneten Datenmuster "X", "S" das Ende der codierten Daten

46a an und zeigen außerdem den Anfang der codierten Daten **46b** an. Gleichermaßen zeigen die mit **48b**, **49b** bezeichneten Datenmuster "X", "S" das Ende der codierten Daten **46b** und den Anfang der codierten Daten **46c** an. Somit können die Datenmuster "X", "S", die im wesentlichen redundant wären, bei einer solchen kontinuierlichen Datenübertragung weggelassen werden.

[0031] Dementsprechend können die Datenabschnitte von variierender Länge sein, wie in den Fig. 2A bis Fig. 2C gezeigt ist. Da die Datenabschnitte von Datenpaketen natürlicherweise in ihren Längen variieren, brauchen sie nicht länger in Rahmen von fester Größe gepresst werden. Wenn beispielsweise ein kurzes Datenpaket zu versenden ist, braucht kein Datenrahmen von fester Größe gesendet zu werden, der größtenteils leere Daten enthält. Vielmehr wird das Ende des Datenpaketes ohne Weiteres gefunden, indem erneut das bekannte Datenmuster "X" erkannt wird.

[0032] Beim Senden von Datenpaketen, wie beispielsweise jene aus den Fig. 2A bis Fig. 2C, über eine Umgebung mit geringem Störabstand wie beispielsweise Luft, ist es möglich, für die Datenmuster "X" und "Z" andere Modulationsregimes zu verwenden als für den Rest des Datenpaketes. Beispielsweise verwenden die Datenmuster, die zur Markierung des Anfangs und des Endes eines Datenpaketes verwendet werden, BPSK-Modulation (Binary Phase Shift Keying), während der Datenabschnitt je nach Nutzerstandort und Funkumgebung beispielsweise mittels 8-PSK (Phase Shift Keying) oder 16-QAM (Quadrative Amplitude Modulation) übertragen wird. Ungeachtet des Modulationsregimes, das für ein bestimmtes Datenpaket verwendet wird, wäre die Modulation, die für die Datenmuster "X" und "Z" verwendet wird, die den Anfang und das Ende eines Datenpaketes anzeigen, in der Regel unveränderlich, um eine Vorrichtung auf einer Empfangsseite zum Erkennen der Datenmuster zu vereinfachen.

[0033] Ein weiterer Aspekt der vorliegenden Erfindung widmet sich der Tatsache, dass das Datenmuster "Z" zufällig in dem Datenabschnitt vorkommen kann, wie beispielsweise in den Datenabschnitten **24**, **34** und **46a**–c in den Fig. 2A bis Fig. 2C. (Dabei wird angenommen, dass das Datenmuster "Z" ein Teilmuster des Musters "X" ist, dergestalt, dass, wenn "Z" substituiert werden, "X" nicht separat substituiert werden müssen, da das Substitut der "Z" alle "X" in den codierten Daten löscht. Wenn allerdings "Z" kein Teilmuster von "X" ist, so kann "X" in der gleichen Weise substituiert werden, die im Weiteren bezüglich "Z" beschrieben wird.) Dieses potenzielle Problem wird in der vorliegenden Erfindung dadurch gelöst, dass das Datenmuster "Z" in den codierten Daten durch ein zweites Datenmuster substituiert wird, wenn bestimmt wird, dass die codierten Daten das Datenmuster "Z" enthalten. Das wird beispielsweise im Zusammenhang mit Fig. 3 beschrieben.

[0034] Fig. 3 veranschaulicht codierte Daten, die ein erstes Datenmuster **102**, das mit "Y" bezeichnet ist, und ein zweites Datenmuster **104**, das mit "Z" bezeichnet ist, enthalten. Somit ist das Datenmuster "Z", das als das bekannte Datenmuster gewählt wurde, welches das Ende eines Datenpaketes bezeichnet, innerhalb der codierten Daten selbst vorhanden. In dem Bestreben, eine falsche Bestimmung zu vermeiden, dass das Datenmuster "Z" ein Indikator für das Ende eines Datenpaketes sei, wird an dessen Stelle ein neues Datenmuster gesetzt, bevor die Daten gesendet werden.

[0035] Genauer gesagt, hat ein Datenmuster "Z" eine Länge l_Z. Wenn dieses Datenmuster "Z" innerhalb der codierten Daten erkannt wird, so wird an dessen Stelle ein zweites Datenmuster mit einem ersten Abschnitt "E" mit einer vorgegebenen Länge l_E verwendet. Ähnlich dem ersten Datenmuster "Z" ist das zweite Datenmuster, das den ersten Abschnitt "E" enthält, eines, das entsprechenden Codierer/Sendern und Empfänger/Decodierern des Systems bekannt sein muss, aber nur von irgend einem bekannten Muster und irgend einer bekannten Länge sein muss. Der erste Abschnitt "E" des zweiten Datenmusters, das in Fig. 3 durch das Element **130** bezeichnet wird, identifiziert, dass eine Substituierung in ursprünglich codierten Daten vorgenommen wurde.

[0036] Wie oben angesprochen, enthält das zweite Datenmuster, das an die Stelle des Datenmusters "Z" gesetzt wurde, einen ersten Abschnitt "E", und enthält des Weiteren einen zweiten Abschnitt "I", der eine Datenart identifiziert, die durch die Substituierung gelöscht wurde. Dieser zweite Abschnitt des Datenmusters ist ein Indikatorfeld mit einer Länge l_I und der Bezeichnung I1 und ist in Fig. 3 als Element **132** gezeigt. Obgleich jeder zweite Abschnitt "I" ein Indikatorfeld ist, das vorzugsweise ein Datenmuster anzeigt, das in ursprünglich codierten Daten substituiert wurde, braucht der zweite Abschnitt "I" nur einer zu sein, der entsprechenden Codierer/Sendern und Empfänger/Decodierern bekannt ist und bekannte Muster und/oder Längen aufweist.

[0037] Wenn also, wie in Fig. 3 gezeigt, das vorgegebene Datenmuster "Z", das dafür verwendet werden soll, das Ende eines codierten Datenpaketes anzuzeigen, in ursprünglich codierten Daten erscheint, so wird es durch ein zweites Datenmuster ersetzt, das einen ersten Abschnitt "E" und einen zweiten Abschnitt "I" enthält. Ein spezieller zweiter Abschnitt "I", wie beispielsweise "I1" (Element **132** von Fig. 3), identifiziert, dass "Z" das Datenmuster war, das in den ursprünglich codierten Daten substituiert wurde. wenn also das ursprüngliche Datenmuster "Z" in den

ursprünglich codierten Daten erscheint, was durch das Element **104** von Fig. 3 angezeigt ist, so wird an dessen Stelle ein zweites Muster gesetzt, das einen ersten Abschnitt "E", der in Fig. 3 durch das Element **130** angezeigt ist, und einen zweiten Abschnitt "I1", der durch das Element **132** angezeigt ist, enthält. Wenn die Daten zum Versenden vorbereitet werden, so werden die Datenmuster "X" und "S" zuerst dem Anfang der codierten Daten (im Anschluss an die oben angesprochene Substituierung) hinzugefügt, was durch die Elemente **120** und **121** angezeigt ist, und das Datenmuster "Z" wird dem Ende der codierten Daten hinzugefügt, was durch das Element **122** angezeigt ist. Anschließend wird dieses codierte Datenmuster von variabler Länge gesendet.

[0038] Vorzugsweise werden bei der eigentlichen Datenübertragung die Datenmuster "X" und "S" praktisch den codierten Daten nach der Substituierung hinzugefügt und zuerst – von den codierten Daten getrennt – gesendet. Danach werden die Datenabschnitte "Y", "E" und "I1" gesendet. Schließlich wird praktisch das Datenmuster "Z" hinzugefügt und gesendet, was das Ende des Datenpaketes bezeichnet.

[0039] Des Weiteren ist es bei einer weiteren bevorzugten Ausführungsform zweckmäßig, wenn der Sender und der Empfänger nicht alle vorkommenden Datenmuster "Z" ersetzen müssen. Darum versteht es sich, dass die Erfindung in einem sequenzsynchronisierten Modus arbeiten kann, wo die codierten Daten in Stücke mit der Länge l_Z segmentiert sind. Jedes Stück wird dann auf das Datenmuster "Z" überprüft und wird anschließend gesendet, anstatt dass auf alle Verschiebungen der Länge l_Z geprüft wird.

[0040] Wie in Fig. 3 gezeigt, ist das, was empfangen wird, ein Datenmuster (vorzugsweise als Abschnitte "X", "S", "Y" usw. empfangen), welches das gleiche ist wie das gesendete, einschließlich des Datenmusters "X", das einen Anfang des Datenpaketes anzeigt, wie durch das Element **120** gezeigt, des Synchronisationsmusters "S" (**121**) und wenigstens des Datenmusters "Z", welches das Ende des Datenpaketes anzeigt, wie durch das Element **122** gezeigt, zusammen mit den codierten Daten "Y", wie durch das Element **122** gezeigt, und dem ersten ("E") und dem zweiten ("I1") Abschnitt des zweiten Datenmusters, das an die Stelle des Datenmusters "Z" gesetzt wurde, wie durch die Elemente **130** und **132** gezeigt. Beim Decodieren der codierten Daten zeigt der erste Abschnitt des zweiten Datenmusters, der durch das Element "E" dargestellt ist, das in Fig. 3 mit **130** bezeichnet ist, an, dass eine Substituierung vorgenommen wurde. Das Indikatorfeld I1, das durch das Element **132** angedeutet ist, zeigt das spezielle Datenmuster an, das substituiert wurde, und zwar in diesem Fall das Datenmuster "Z". Dementsprechend werden unter Verwendung dieser Informationen der erste und der zweite Abschnitt des zweiten Datenmusters dann durch das Datenmuster "Z" ersetzt, was durch das Element **150** von Fig. 3 angedeutet ist, um die Wiederherstellung der korrekt codierten Daten zu ermöglichen.

[0041] Ähnlich den Problemen, zu denen es kommen kann, wenn das Datenmuster "Z" in den codierten Daten vorkommt, weil es als ein Indikator des Endes eines codierten Datenpaketes verwendet wird (und vorzugsweise als ein Abschnitt des Datenmusters "X" verwendet wird, der den Anfang eines codierten Datenpaketes anzeigt, unter der Annahme, dass "Z" ein Teilmuster oder eine Teilmenge von "X" ist), kann es auch zu einem Problem kommen, wenn das Datenmuster "E" in den ursprünglich codierten Daten vorkommt. Genauer gesagt, weil das Datenmuster "E" verwendet wird, um zu identifizieren, dass eine Substituierung vorgenommen wurde, zeigt das Datenmuster "E", wenn es in den ursprünglich codierten Daten vorkommt, fälschlicherweise an, dass eine Substituierung vorgenommen wurde. Um ein solches mögliches Problem zu lösen, wird das Datenmuster "E", wenn es in den ursprünglich codierten Daten vorkommt, vorzugsweise ebenfalls durch ein zweites Datenmuster ersetzt, wobei dieses zweite Datenmuster einen gleichen Abschnitt "E", der anzeigt, dass eine Substituierung vorgenommen wurde, und einen anderen zweiten Abschnitt "I" enthält. Dieser zweite Abschnitt "I" zeigt an, dass das Datenmuster "E" in den ursprünglich codierten Daten vorhanden war, und sich darum von dem zweiten Abschnitt "I1" unterscheidet, der verwendet wird, um das Datenmuster "Z" anzuzeigen. Dementsprechend wird ein anderer zweiter Abschnitt I2 des zweiten Datenmusters, der in Fig. 3 nicht gezeigt ist, verwendet, um das Vorhandensein des Datenmusters "E" in den ursprünglich codierten Daten anzuzeigen. Vorzugsweise wird der zweite Abschnitt "I" des zweiten Datenmusters, der die Art von Daten identifiziert, die ursprünglich in den codierten Daten vorhanden waren, durch ein Datenmuster identifiziert (wobei das Datenmuster für das Feld "I", welches das Datenmuster "Z" darstellt, ein anderes ist als für das Feld "I2", welches das Datenmuster "E" darstellt). Fig. 4 veranschaulicht eine Vorrichtung **200** zum Codieren von Daten und zum Senden von codierten Daten mit variabler Länge in einer Umgebung mit geringem Störabstand. Die Vorrichtung **200** von Fig. 4 wird im Anschluss erklärt. Es ist zu beachten, dass viele Beispiele für eine drahtlose Signalübertragung als Umgebung mit geringem Störabstand genannt werden, aber die Erfindung ist nicht auf eine bestimmte Umgebung mit geringem Störabstand beschränkt. Des Weiteren beinhalten viele genannte Beispiele CDMA-Anwendungen, aber die längenvariablen Paketierungsregime-Anwendungen der vorliegenden Anmeldung gelten gleichermaßen auch für andere Anwendungen, wie beispielsweise Vielfachzugriff im Zeitmultiplex (TDMA).

[0042] Die Daten **202** werden zunächst vorzugsweise in den Datenpuffer **203** eingespeist. Ein den Pufferinhalt anzeigendes Signal wird an die Steuerung **208** gesendet. Wenn der Puffer **203** voll ist, so werden die Daten **202** zum Codierer **204** übermittelt. Natürlich ist zu beachten, dass eine solche Pufferung von Daten auch lediglich durch die Steuerung **208** gesteuert werden könnte und/oder im Codierer **204** stattfinden könnte, wodurch die Notwendigkeit eines separaten Datenpuffers **203** entfällt.

[0043] Wenn die Umgebung mit geringem Störabstand eine drahtlose Umgebung wie beispielsweise Luft ist, so handelt es sich bei dem Codierer **204** vorzugsweise beispielsweise um einen Vorwärtsfehlerkorrekturcodierer. In einer Umgebung mit geringem Störabstand, wie beispielsweise einer drahtlosen CDMA-Umgebung, wird der Vorwärtsfehlerkorrekturcodierer dafür verwendet, Daten redundant zu codieren, was auf ein natürlich auftretendes Problem mit Fehlern zurückzuführen ist, die sich aus dem Umgebung mit geringem Störabstand ergeben. Wenn ein Datenmuster aus "N" Bits besteht, so gibt der FEC-Codierer dementsprechend 2N oder 3N Bits an Daten aus.

[0044] An den Codierer **204** schließt sich vorzugsweise eine Verschachtelungsvorrichtung **206** an. Die Verschachtelungsvorrichtung **206** dient in einem drahtlosen System wie beispielsweise einem CDMA-System dazu, die Leistung eines FEC-Codierers zu verbessern, indem die Auswirkungen von Mehrfachfehlern in einer Zeile verringert werden. Die codierten Daten werden praktisch durch die Verschachtelungsvorrichtung **206** gemischt, und/oder Bits werden durch die Verschachtelungsvorrichtung **206** verschoben. Die Verwendung einer Verschachtelungsvorrichtung **206** ist in der drahtlosen Kommunikation üblich, so dass nicht näher darauf eingegangen wird.

[0045] Der Codierer **204** und die Verschachtelungsvorrichtung **206** werden durch eine Steuerung **208** gesteuert. Die Steuerung **208** beginnt ihre Steuerung, wenn ein Eingabepuffer voll ist, und synchronisiert den Codierer **204** mit einer Paketgrenze der Daten **202**.

[0046] An die Verschachtelungsvorrichtung **206** schließt sich ein Multiplexer **210** an. Der Multiplexer **210** multiplext in den Datenausgang von der Verschachtelungsvorrichtung **206** alle N Bits, wie beispielsweise alle 10 Bits, Leistungssteuerungsbits hinein. Diese Leistungssteuerungsbits können durch Überschreiben anderer Bits in einer bekannten Weise eingefügt werden, und die LS-Bits sagen anderen Sendern, wie die Leistung zu steuern ist.

[0047] Die Steuerung **208** veranlasst den Multiplexer **210**, Leistungssteuerungsbits in den codierten Datenausgang von der Verschachtelungsvorrichtung **206** einzufügen, und steuert des Weiteren, dass die vorgegebenen Datenmuster "X", "S" und "Z" in den Anfang und das Ende der codierten Daten gemultiplext werden, wie weiter unten noch näher erläutert wird.

[0048] An den Multiplexer **210** schließt sich ein Konverter **220** an, dessen Funktion eingehend im Zusammenhang mit der Funktionsweise des Systems erläutert wird. Kurz gesagt, untersucht der Konverter **220** die Daten selbst, um zu bestimmen, ob das Datenmuster "Z" in den codierten Daten vorhanden ist oder nicht, und substituiert es durch ein zweites Datenmuster (das beispielsweise einen ersten Abschnitt "E" und einen zweiten Abschnitt "I" enthält). Dieser Konverter **220** kann beispielsweise einen Speicher enthalten, der Informationen über das Datenmuster "Z" beinhaltet, so dass er das bekannte Datenmuster "Z" mit den codierten Daten vergleichen kann, um das Vorhandensein oder das Fehlen des Datenmusters "Z" zu bestimmen. Des Weiteren speichert ein solcher Speicher vorzugsweise das zusätzliche Datenmuster "E", da die codierten Daten ebenfalls vorzugsweise auf das Vorhandensein des Musters E untersucht werden. Außerdem enthält der Speicher des Weiteren alle Substitut-Datenmuster, einschließlich des ersten Abschnitts E und der verschiedenen zweiten Abschnitte I_Z, I_E usw. Es ist zu beachten, dass der Konverter **220** nicht nur auf Datenmuster prüft, die in den codierten Daten vorhanden sind, sondern auch dazu dient, erforderlichenfalls ein entdecktes Datenmuster durch einen neues Datenmuster zu substituieren oder zu ersetzen.

[0049] An den Konverter **220** schließt sich ein Multiplexer **230** an. Der Multiplexer **230** wird durch die Steuerung **208** dahingehend gesteuert, ein vorgegebenes Datenmuster "X" und ein Synchronisationsmuster "S" an den Anfang eines codierten Datenausgangs vom Konverter **220**, einschließlich aller daran vorgenommenen Substituierungen, anzufügen und ein vorgegebenes Datenmuster "Z" an ein Ende des codierten Datenausgangs anzufügen. Das Datenmuster "X" wird, zusammen mit dem Datenmuster "S", am Anfang der codierten Daten eingefügt, um den Anfang der codierten Daten anzuzeigen; und das Datenmuster "Z" wird am Endteil der codierten Daten angefügt, um das Ende der codierten Daten anzuzeigen. Fig. 4 veranschaulicht des Weiteren einen Spreizer **240**, der den Signalausgang vom Multiplexer **230** in bekannter Weise spreizt. Zu weiteren, nicht gezeigten Komponenten können ein Datenmodulator und ein Leistungsverstärker gehören, die vorzugsweise zwischen dem Spreizer **240** und einer Antenne **250** angeordnet sind.

[0050] Schließlich wird der Signalausgang vom Spreizer **240** über die Antenne **250** ausgegeben und als ein ausgebreitetes Signal **260** in eine Umgebung mit geringem Störabstand, wie beispielsweise Luft,

gesendet. Der Spreizer **240** und die Antenne **250** sind in Fig. 4 so dargestellt, dass sie in der Vorrichtung **200** enthalten sind. Das ist aber nicht als Einschränkung zu verstehen, da auch ein Spreizer und/oder eine Antenne außerhalb der Vorrichtung **200** zur Signalübertragung verwendet werden können.

[0051] Das ausgebreitete Signal **260**, das in einem Medium mit geringem Störabstand, wie beispielsweise Luft, verkörpert ist, enthält ein Anfangssegment, das ein Datenmuster "X" enthält, welches einen Anfang des ausgebreiteten Signals identifiziert, ein Datenmuster "S" und ein Endsegment, welches das Datenmuster "Z" enthält, welches ein Ende des ausgebreiteten Signals identifiziert. Des Weiteren ist ein codiertes Datensegment darin enthalten. Wenn das codierte Datensegment ursprünglich das Datenmuster "Z" enthielt, so wird vor der Signalausbreitung ein anderes Datenmuster an dessen Stelle gesetzt. Bei einer bevorzugten Ausführungsform ist das ausgebreitete Signal **260** beispielsweise in einer Trägerwelle verkörpert, die durch die Luft wandert.

[0052] Fig. 5 veranschaulicht eine Decodierungsvorrichtung **300**. Ähnlich der Vorrichtung **200**, beinhaltet die Beschreibung der Decodierungsvorrichtung **300** viele Verweise auf drahtlose Signalübertragung, ist aber nicht darauf beschränkt.

[0053] Wie in Fig. 5 gezeigt, wird das ausgebreitete Signal **260** durch eine Antenne **310** empfangen. Auch hier braucht eine solche Antenne (und der Entspreizer **340**) nicht in der Decodierungsvorrichtung **300** enthalten zu sein, sondern kann sich außerhalb der Decodierungsvorrichtung **300** befinden. Daran schließt sich ein Detektor **330** an, der in einem drahtlosen System vorzugsweise ein Korrelator ist. Der Korrelator, vorzugsweise mit Teilchipabtastung, wird an das Datenmuster "X" angepasst. Der Detektor **330** oder Korrelator ist darum dafür konfiguriert, die Anfangsabschnitte eines empfangenen codierten Signals mit variabler Länge zu erkennen, indem er beispielsweise ein Datenmuster "X" erkennt. Nachdem dieses Datenmuster erkannt wurde, erfolgt eine Suche nach dem Datenmuster "S", woraufhin ein Abschnitt codierter Daten identifiziert wird. Zu weiteren, nicht gezeigten Komponenten können ein Frontend-Empfänger und eine Zeitsteuerungsschaltung (die einen Teil des Detektors **330** bilden könnte) und ein Datenmodulator, der zwischen dem Detektor **330** und dem Entspreizer **340** angeordnet ist, gehören.

[0054] Anschließend wird der Signalausgang vom Detektor **330** in den Entspreizer **340** eingegeben, um das Signal auf bekannte Weise zu entspreizen. Daran schließt sich ein Konverter **350** an, der dafür konfiguriert ist zu bestimmen, ob ein Abschnitt des erkannten empfangenen codierten Datensignals mit variabler Länge ein zweites Datenmuster enthält, das im Anschluss an die Datenmuster "X" und "S" einen ersten Abschnitt "E" enthält. Der erste Abschnitt "E" des zweiten Datenmusters dient dazu, eventuelle Substituierungen zu identifizieren, die in dem codierten Datensignal vorgenommen wurden. Wenn ein solches zweites Datenmuster in dem empfangenen codierten Datensignal mit variabler Länge enthalten ist, so wird an dessen Stelle ein neues Datenmuster gesetzt. Ähnlich dem Konverter **220** enthält der Konverter **350** vorzugsweise einen Speicher, der sämtliche vorgegebenen Datenmuster, einschließlich deren erster und zweiter Abschnitte, speichert.

[0055] Wie im Zusammenhang mit dem Systembetrieb eingehend erläutert werden wird, wird – sobald das Datenmuster "E" innerhalb des empfangenen codierten Datensignals mit variabler Länge entdeckt wird, was anzeigt, dass eine Substituierung vorgenommen wurde – ein nachfolgender Abschnitt des codierten Datensignals mit variabler Länge untersucht, um zu bestimmen, welches Muster "I" danach erscheint. Je nach dem Muster "I" wird das gesamte zweite Datenmuster, einschließlich des ersten Abschnitts "E" und des zweiten Abschnitts "I", durch eines von mehreren Datenmustern substituiert. Wenn beispielsweise das Datenmuster "Z" (**104**) ursprünglich in den codierten Daten vorhanden ist, so ist das zweite Muster, einschließlich des ersten Abschnitts "E" (**130**) und des zweiten Abschnitts "I" (**132**), in dem empfangenen Datensignal vorhanden, und das Datenmuster "Z" (**150**) wird durch den Konverter **350** an dessen Stelle gesetzt.

[0056] Danach schließt sich an den Konverter **350** ein Demultiplexer **360** an, der die Leistungssteuerungsbits aus den codierten Daten heraustrennt. Diese Steuerungsbits LS lassen sich leicht heraustrennen, da sie uncodiert übertragen werden und da sie in regelmäßigen Intervallen innerhalb der codierten Daten vorkommen. An den Demultiplexer **360** schließt sich eine Entschachtelungsvorrichtung **370** an. Schließlich schließt sich an die Entschachtelungsvorrichtung **370** ein Decodierer **380** an, der das codierte Datensignal decodiert, das durch den Konverter **350** modifiziert wurde. Dieses Decodieren erfolgt beispielsweise nach den erkannten Datenmustern "X" und "S". Der Decodierer **380** ist vorzugsweise ein FEC-Decodierer. Da solche Decodierer bekannt sind, wird auf den FEC-Decodierer nicht weiter eingegangen.

[0057] In der Decodierungsvorrichtung **300** ist des Weiteren eine Steuerung **390** enthalten, die den Demultiplexer **360**, den Detektor **330**, den Konverter **350**, die Entschachtelungsvorrichtung **370** und den Decodierer **380** steuert. Vom Decodierer **380** werden Daten **392** ausgegeben, wobei diese Daten jenen der Eingangsdaten **202** von Fig. 4 äquivalent sind.

[0058] Außerdem kann die Decodierungsvorrich-

tung **300** des Weiteren eine Schlummer-/Weckschaltung **320** enthalten, die mit der Antenne **310** und dem Detektor **330** verbunden ist. Während langer Zeiträume, in denen keine Daten empfangen werden, kann die Decodierungsvorrichtung **300** in einen Schlummermodus umschalten, um die Batterie zu schonen. Die Schlummer-/Weckschaltung **320** bewirkt dann eine Reaktivierung der Decodierungsvorrichtung **300** in Reaktion beispielsweise auf den Empfang eines Rufsignals, das durch die Antenne **310** empfangen wird, oder beispielsweise auf den Empfang von Energie, die auf einem Verkehrskanal empfangen wird.

[0059] Die Fig. 6 und Fig. 7 beschreiben kurz das Gesamtverfahren zum Senden und Empfangen von codierten Daten mit variabler Länge in einer Umgebung mit geringem Störabstand. Fig. 6 veranschaulicht die Datenübertragung, und Fig. 7 veranschaulicht das Empfangen und Decodieren von Daten.

[0060] Zunächst, wie in Fig. 6 gezeigt, werden in Schritt S2 Eingangsdaten aufgefüllt (beispielsweise durch FEC-Codierung in CDMA) und codiert. Danach werden diese Daten in Schritt S4 gemischt oder verschachtelt. In Schritt S6, wobei eine separate Übertragung von Datenabschnitten angenommen wird, wird durch den Multiplexer **230** ein Datenmuster "X" (zusammen mit einem Synchronisationsdatenmuster "S") hinzugefügt, und die Datenmuster "X" und "S" werden gesendet. Es ist anzumerken, dass ein bevorzugtes Verfahren des Sendens von Abschnitten eines Datenpaketes, wie beispielsweise jenes, das in Fig. 2C gezeigt ist, in Bezug auf Fig. 6 besprochen wird. Dies gestattet das häufige Einfügen, Zuweisen und Senden von LS-Bits. Jedoch ist die Erfindung nicht darauf beschränkt und kann die Paketierung eines gesamten Datenpaketes, wie beispielsweise jenes, das in Fig. 2C gezeigt ist, vor dem Senden sowie das Senden eines kompletten Datenpaketes, einschließlich der Anfangs- und End-Datenmuster "X" und "Z", enthalten.

[0061] Danach wird in Schritt S8 bestimmt, ob alle codierten Daten gesendet wurden oder nicht. Wenn nicht, wird ein Teil der codierten Daten in Schritt S10 untersucht. Dabei wird angenommen, dass das codierte Datensignal beispielsweise unter Verwendung des Datenpuffers **203** so aufgeteilt oder segmentiert wurde, dass nur Teile des codierten Datensignals auf das Vorhandensein von vorgegebenen Mustern untersucht zu werden brauchen. Erforderlichenfalls werden in Schritt S12 LS eingefügt.

[0062] Als nächstes wird in Schritt S14 bestimmt, ob das codierte Signal ein bestimmtes Datenmuster enthält, wie beispielsweise das vorgegebene Datenmuster "Z". Wenn das codierte Signal ein solches bestimmtes Datenmuster enthält, so wird dieses Datenmuster in Schritt S16 durch ein anderes Datenmuster ersetzt. Beispielsweise wird das Datenmuster "Z", wenn es in den codierten Daten vorhanden ist, durch ein Datenmuster ersetzt, das beispielsweise einen ersten Abschnitt "E" und einen zweiten Abschnitt "I" enthält. Wenn die Antwort auf Schritt S14 NEIN ist, so schreitet das Verfahren zu Schritt S18 voran. Gleichermaßen schreitet das System auch zu S18 voran, nachdem das bestimmte Datenmuster in Schritt S6 durch ein anderes Datenmuster ersetzt wurde. Bei Schritt S18 wird der codierte Datenteil gesendet, wenn er nicht das bestimmte Datenmuster (von Schritt S14) enthält, und der codierte Datenteil wird einschließlich der ersetzten Datenmuster gesendet, wenn das codierte Signal ursprünglich das bestimmte Datenmuster (von Schritt S16) enthielt. Darum kann der codierte Datenteil, der in Schritt S18 gesendet wurde, beispielsweise ein Substitut-Datenmuster enthalten, das einen ersten Abschnitt "E", wie durch das Element **130** in Fig. 3 angedeutet, und einen zweiten Abschnitt "I1", wie durch das Element **132** in Fig. 3 angedeutet, enthalten.

[0063] Schließlich, nachdem die Datenübertragung in Schritt S18 vollständig ist, kehrt das System zu Schritt S8 zurück, um zu untersuchen, ob das Ende der codierten Daten erreicht ist oder nicht. Wenn es noch weitere codierte Datenteile gibt, so schreitet das Verfahren zu Schritt S10 voran. Wenn das Ende der codierten Daten in Schritt S8 erreicht ist, so schreitet das System zu Schritt S20 voran. In Schritt S20 wird ein vorgegebenes Postambel-Muster "Z" gesendet, um das Ende des codierten Datensignals anzuzeigen. Auch hier ist anzumerken, dass die Erfindung das Zusammenstellen und Senden eines kompletten Datenpaketes, wie es beispielsweise in Fig. 2A gezeigt ist, umfasst, wobei die Präambel- und Postambel-Datenabschnitte "X" und "Z" (und das Synchronisationsmuster "S") dem codierten Datenmuster vor der Übertragung angefügt werden.

[0064] Fig. 7 erklärt kurz, wie codierte Daten mit variabler Länge empfangen und decodiert werden. In Schritt S50 wird ein Signal empfangen. Danach wird das Signal in Schritt S52 untersucht, um zu bestimmen, ob das bestimmte Datenmuster "X" erkannt wurde. Wenn kein solches Datenmuster "X" erkannt wird, so kehrt das System zu Schritt S50 zurück und wartet auf den Empfang eines weiteren Signals. Wenn jedoch das bestimmte Datenmuster "X" in S52 erkannt wird, so wird das empfangene Signal auf das Vorhandensein des Datenmusters "S" untersucht. Ist das der Fall, so zeigt dies an, dass das Signal nach dem Datenmuster "S", das zur Synchronisierung der Decodierungsvorrichtung **300** verwendet wird, das codierte Signal ist, und das Signal wird bei Schritt S54 weiter untersucht.

[0065] In Schritt S56 wird bestimmt, ob ein anderes bestimmtes Datenmuster erkannt wurde. Ist das der Fall, so geht das Verfahren zu Schritt S58 über, um zu bestimmen, ob das bestimmte Datenmuster ein

"Z"-Muster ist, was das Ende eines Datenpaketes anzeigt.

[0066] Wenn das bestimmte Datenmuster in Schritt S58 nicht das "Z"-Muster ist, so geht das Verfahren zu Schritt S62 über. Wenn in Schritt S56 ein anderes bestimmtes Datenmuster, wie beispielsweise das Datenmuster, das einen ersten Abschnitt "E" enthält, erkannt wird, so wird das gesamte Datenmuster, einschließlich eines ersten Abschnitts "E" und eines zweiten Abschnitts "I", in Schritt S62 durch ein anderes Datenmuster ersetzt, was sich nach dem bestimmten zweiten Abschnitt "I" richtet. Wenn beispielsweise, wie in Fig. 3 gezeigt, das Datenmuster "E", das durch das Element **130** angedeutet ist, und das Datenmuster "I1", das durch das Element **132** angedeutet ist, erkannt werden, so wird an deren Stelle das Datenmuster "Z", das durch das Element **150** angedeutet ist, gesetzt. Danach schreitet das Verfahren zu Schritt S64 weiter. Gleichermaßen schreitet das Verfahren zu Schritt S64 voran, wenn in Schritt S56 kein bestimmtes Datenmuster erkannt wurde.

[0067] In Schritt S64 werden, sofern zutreffend, LS-Bits extrahiert, und das Verfahren kehrt zu Schritt S54 zurück, wo das eingehende Signal untersucht wird. Das Verfahren wird wiederholt, bis in den Schritten S56 und S58 bestimmt wird, dass ein Datenmuster "Z" erkannt wurde, was das Ende des codierten Datensignals anzeigt. Danach findet in Schritt S66 die Entschachtelung statt. Danach findet in Schritt S68 das Decodieren des codierten Datensignals statt. wenn das der Fall ist, endet das Verfahren mit Schritt S60.

[0068] Dementsprechend wird durch die Vorrichtung und das Verfahren der vorliegenden Erfindung ein Datenmuster bestimmt, das der Vorrichtung **200** und der Decodierungsvorrichtung **300** bekannt bleibt. Dieses Datenmuster "X" wird anschließend verwendet, um ein eingehendes codiertes Datensignal mit variabler Länge anzuzeigen und eine Zeitsteuerungssynchronisation zu erreichen, um das Erkennen des Anfangs des codierten Datensignals von variabler Länge (mit dem Datenmuster "S") zu ermöglichen. Ein weiteres Datenmuster "Z", welches das gleiche wie das Datenmuster "X" oder eine Teilmenge des Datenmusters "X" sein kann, wird dafür verwendet, ein Ende von codierten Daten mit variabler Länge zu bestimmen. Die bekannten Datenmuster "X" und "Z" müssen in der Regel länger sein als jenes, das normalerweise in verdrahteten Anwendungen verwendet wird, damit die codierten Daten mit variabler Länge in einer Umgebung mit geringem Störabstand, wie beispielsweise dem Störabstand von Luft, die in der drahtlosen Kommunikation benutzt wird, übertragen werden können. Dies ist notwendig, um die von geringem Störabstand geprägte Situation, Schwund usw., wozu es normalerweise in einem drahtlosen Kanal kommt, zu überwinden. Des Weiteren kann die vorliegende Erfindung, obgleich sie bevorzugt in der drahtlosen Kommunikation verwendet wird, auch in verdrahteten Anwendungen verwendet werden, insbesondere solchen mit geringem Störabstand.

[0069] Bei einer weiteren bevorzugten Ausführungsform der vorliegenden Anmeldung werden weitere Datenmuster, die dem Datenmuster "E" und dem Datenmuster "Z" ähneln, vorzugsweise ebenfalls ersetzt, wenn sie auf natürliche Weise in den codierten Daten vorkommen. Fig. 8 veranschaulicht ein Beispiel, wo nicht nur das Datenmuster "Z", das dafür verwendet wird, das Ende von codierten Daten mit variabler Länge anzuzeigen, und das Datenmuster "E", das eine Substituierung von Daten anzeigt, sondern auch ein Datenmuster Z', das dem Datenmuster "Z" ähnelt, ersetzt werden müssen.

[0070] Bei dem in Fig. 8 gezeigten Beispiel enthalten codierte Daten einen ersten Abschnitt "Y", der durch das Element **102** bezeichnet ist; einen zweiten Abschnitt "Z", der durch das Element **104** bezeichnet ist; einen dritten Abschnitt Z', der durch das Element **106** bezeichnet ist, und einen vierten Abschnitt "Y", der durch das Element **108** bezeichnet ist. Bei der Vorbereitung der Daten auf die Übertragung werden die vorbestimmten Datenmuster "X" (**120**) und "S" (**121**) an den Anfang der codierten Daten angefügt, und das Datenmuster "Z" (**122**) wird dem Ende der codierten Daten angefügt. Das Datenmuster "Z", das durch das Element **104** bezeichnet wird, wird substituiert, wobei ein Datenmuster, das einen ersten Abschnitt "E", der durch das Element **130** angedeutet ist, und einen zweiten Abschnitt I1, der durch das Element **132** angedeutet ist, enthält, hergestellt wird. Jedoch wird bei dem in Fig. 8 gezeigten Beispiel das Muster "Z'", das durch das Element **106** angedeutet ist, nicht substituiert.

[0071] Wenn das codierte Datensignal empfangen wird, so wird das Muster "X", das durch das Element **120** angezeigt ist, zunächst erkannt, wie auch das Synchronisationsmuster "S" (**121**). Danach wird das codierte Datensignal gelesen und festgestellt, dass es das Muster "Y", das durch das Element **102** angedeutet ist, und das Datenmuster, das einen ersten Abschnitt "E", der durch das Element **130** angedeutet ist, und einen zweiten Abschnitt "I1", der durch das Element **132** angedeutet ist, beinhaltet. Das Muster "Y" wird aus dem Abschnitt "E" decodiert, das durch das Element **102** angedeutet ist, und das Muster "Z" wird aus dem Abschnitt "I1" decodiert, das durch das Element **150** angedeutet ist. Weil aber "Z'" dem Datenmuster "Z" sehr ähnlich ist, kann es während der Übertragung des Signals zu einem Fehler kommen, wenn das empfangene Signal empfangen wird. Anstatt also zu erkennen, dass ein Muster "Z'" empfangen wird, was durch das Element **106** angedeutet ist,

kann das empfangene Muster irrtümlicherweise als ein weiteres Muster "Z" identifiziert werden, was durch das Element **140** angedeutet ist, und es kann zu einem Fehler kommen.

[0072] Anstatt ein Muster "Z'" zu decodieren, was durch 106 angedeutet ist, und anschließend das Muster "Y" zu decodieren, was durch das Element **108** angedeutet ist, bis ein weiteres Muster "Z", das durch das Element **122** angedeutet ist, erkannt wird – wird also, wie in Fig. 8 gezeigt, ein Paket-Ende signalisiert, da versehentlich ein weiteres Element "Z" erkannt wird, was durch das Element **140** angedeutet ist. Statt dass also das Signal ordnungsgemäß als die Datenmuster "Y", "Z", "Z'" und "Y" enthaltend decodiert wird, wird es als ein Datenmuster "Y, Z" decodiert, was durch die Elemente **102** und **150** von Fig. 8 gezeigt ist, wobei ein zweites Datenmuster als ein Datenmuster "Y" decodiert wird, was durch das Element **108** von Fig. 8 gezeigt ist. Dazu kann es kommen, weil das empfangene Datensignal möglicherweise so wahrgenommen wird, als enthielte es ein erstes Muster "X", was durch das Element **120** angedeutet ist und den Anfang eines Datenmusters anzeigt, ein zweites Datenmuster "Z", was durch das Element **140** angedeutet ist und ein Ende eines ersten codierten Datensignals anzeigt, und ein weiteres Datenmuster "Z", was durch das Element **122** angedeutet ist und das Ende eines zweiten Datenmusters anzeigt. Auch hier kann dies geschehen, weil das ursprünglich codierte Datenmuster **106** ein Muster enthält, das einem Datenmuster "Z" gleicht oder stark ähnelt.

[0073] Fig. 9 veranschaulicht eine weitere bevorzugte Ausführungsform der vorliegenden Anmeldung, wo nicht nur das Datenmuster "Z" substituiert wird, wenn es auf natürliche Weise in den codierten Daten vorkommt, sondern wo auch das Datenmuster Z', was durch das Element **106** angedeutet ist, substituiert wird. In dieser Situation werden vor der Übertragung der Daten die Datenmuster "X" und "S" (**120**, **121**) an den Anfang der codierten Daten angefügt, und das Datenmuster "Z" (**122**) wird an das Ende der codierten Daten angefügt. Das Datenmuster "Z", das durch das Element **104** angedeutet ist, wird innerhalb der codierten Daten erkannt, und ein Muster mit einem ersten Abschnitt "E", der durch das Element **130** angedeutet ist, und einem zweiten Abschnitt "I1", der durch das Element **132** angedeutet ist, wird an dessen Stelle gesetzt.

[0074] Des weiteren wird innerhalb der codierten Daten erkannt, dass auch ein Datenmuster "Z'", das durch das Element **106** angedeutet ist, vorhanden ist. Dementsprechend wird auch das Datenmuster "Z'", das durch das Element **106** angedeutet ist, ersetzt, wobei ein Datenmuster mit einem ersten Abschnitt "E", der durch das Element **101** angedeutet ist, an dessen Stelle gesetzt wird, zusammen mit einem zweiten Abschnitt I2, der durch das Element **133** angedeutet ist. Dieses Element "I2" ist vorzugsweise von einem anderen Muster als das Element "I1", dergestalt, dass das Element "Z'" nach dem Decodieren an dessen Stelle gesetzt wird, wie durch das Element **151** von Fig. 9 gezeigt. Dementsprechend wird bei einer solchen weiteren bevorzugten Ausführungsform der vorliegenden Erfindung das ursprünglich codierte Datensignal, das die Datenmuster "Y", "Z", "Z'" und "Y" enthält, ordnungsgemäß decodiert und wiederhergestellt, wodurch die in Fig. 8 gezeigten Probleme vermieden werden.

[0075] Bei einer weiteren bevorzugten Ausführungsform werden vorzugsweise nicht nur die Datenmuster "Z", "Z'" und "E" substituiert, wenn sie auf natürliche Weise in einem codierten Datensignal vorkommen, sondern es wird vorzugsweise auch ein Datenmuster "E", das dem Datenmuster E ähnelt, substituiert. Somit wird jedes der vier bekannten Datenmuster durch ein Datenmuster mit einem ersten Abschnitt "E" und durch einen zweiten Abschnitt "I1", "I2", "I3" oder "I4" substituiert. "I1" bis "I4" zeigen jeweils die Art der Daten an, die substituiert wurden. Vorzugsweise identifizieren I1 bis I4 jeweils die Art der Daten, die durch ein Muster substituiert werden. Somit sind I1 bis I4 vorzugsweise jeweils von unterschiedlichem Muster.

[0076] In dem Bestreben, das Decodieren eines falschen Signals zu vermeiden, markieren insbesondere bei dieser weiteren bevorzugten Ausführungsform die Vorrichtung und das Verfahren der vorliegenden Erfindung Varianten oder "nächste Nachbarn" Sequenzen von Datenmustern mit "Z" und "E". Eine benachbarte Gruppe, die mehr als eine Variante von Z enthalten kann, kann als Z_j eingestellt werden, als eine Gruppe von Datenmustern, die sich von Z in j Stellen unterscheiden. Gleichermaßen kann E_j eine Gruppe von Datenmustern darstellen, die sich von E in j Stellen unterscheiden. Somit können "E'" und "Z'" jeweils mehreren Mustern entsprechen, die zu substituieren sind.

[0077] Wenn wir der Einfachheit halber annehmen, dass j = 1, wobei Z = 0101, so gibt es vier benachbarte Datenmuster: 1101, 0001, 0111 und 0100. Wenn eines oder mehrere der vier Datenmuster erkannt werden, so werden sie darum durch ein Datenmuster substituiert, das einen ersten Abschnitt "E" und einen von vier zweiten Abschnitten "I" enthält. Bezüglich E werden vorzugsweise vier ähnliche Datenmuster ersetzt. Jedes Datenmuster wird durch ein Datenmuster ersetzt, das einen ersten Abschnitt "E", gefolgt von einem zweiten Abschnitt "I", enthält, der jedes Muster eindeutig identifiziert. Jedes der Datenmuster, die innerhalb eines codierten Signals zu überprüfen sind, wird vorzugsweise in einem Speicher gespeichert, zusammen mit Mustern, die an deren Stelle treten. Gleichermaßen werden diese Muster eben-

falls vorzugsweise in einem Speicher zum Zweck des Decodierens gespeichert.

[0078] Des Weiteren befassen sich weitere bevorzugte Ausführungsformen der vorliegenden Anmeldung gleichermaßen mit Situationen, wo ein Datenmuster "E" für ein Datenmuster "E'" gehalten werden könnte. In einer solchen Situation wird eine Escape-Sequenz möglicherweise nicht erkannt, und ein codiertes Signal wird möglicherweise nicht ordnungsgemäß decodiert. Schließlich wird ein Datenmuster "E'" möglicherweise für eine Datensequenz E gehalten, was zu einer zusätzlichen Substituierung einer Escape-Sequenz führt.

[0079] Solche möglichen Situationen können gemildert werden, indem eine Gruppe von Escape-Sequenzen ("Z", "E", "Z_j" und "E_j") als eine Funktion von j, mit Sequenzmustern "I_Z", "I_E" und "I_I", richtig ausgewählt wird. Je mehr Sequenzen hinzugefügt werden, desto höher wird zwar die Genauigkeit, aber es steigen möglicherweise auch die Kosten und die Zeit für das Substituieren von Datenmustern beim Senden und Empfangen von Daten. Das heißt, obgleich alle derartigen Aspekte in der vorliegenden Erfindung enthalten sind, müssen doch die praktischen Realisierungsfaktoren bedacht werden, weil Zeit und Kosten gegen die Genauigkeit abgewogen werden müssen.

[0080] Das Feld "I" kann ebenfalls vorzugsweise codiert werden, um beispielsweise den Abstand zwischen zwei Sequenzen "I_1" und "I_2" zu vergrößern. Die Fehlererkennung kann verbessert werden, wenn Datenpakete eines höheren Protokolls übertragen werden, die ein Längenfeld oder eine CRC-Prüfung beinhalten, die dafür verwendet werden kann, eine falsche Operation zu erkennen und zurückzuweisen.

[0081] Dank der Fähigkeit, den Anfang und das Ende von codierten Datensignalen korrekt zu bestimmen, können somit codierte Datensignale mit variabler Länge in Umgebungen mit geringem Störabstand, wie beispielsweise in Umgebungen für die drahtlose Kommunikation, übertragen werden. Des Weiteren können durch das Substituieren bestimmter Muster innerhalb codierter Daten falsche Anzeigen von codierten Datensignalen mit variabler Länge vermieden werden.

Patentansprüche

1. Verfahren zum Codieren von Daten, mit den folgenden Schritten:
Bestimmen, ob codierte Daten ein erstes Datenmuster enthalten oder nicht;
Substituieren des ersten Datenmusters in den codierten Daten durch ein zweites Datenmuster, wenn bestimmt wird, daß die codierten Daten das erste Datenmuster enthalten;
Hinzufügen des ersten Datenmusters zu einem Endteil der codierten Daten, um das Ende der codierten Daten anzuzeigen;
dadurch gekennzeichnet, daß
der Schritt des Bestimmens das Bestimmen umfaßt, ob die codierten Daten eine Variante des ersten Datenmusters enthalten oder nicht und wobei eine bestimmte Variante des ersten Datenmusters in den codierten Daten durch das zweite Datenmuster substituiert wird.

2. Verfahren nach Anspruch 1, ferner mit dem folgenden Schritt:
Übertragen der codierten Daten in drahtloser Kommunikation.

3. Verfahren nach Anspruch 1, ferner mit dem folgenden Schritt:
Codieren von Daten durch Codierung mit Vorwärtsfehlerkorrektur (FEC) vor dem Schritt des Bestimmens.

4. Verfahren nach Anspruch 2, ferner mit dem folgenden Schritt:
Übertragen der codierten Daten von einer drahtlosen Kommunikationseinrichtung.

5. Verfahren nach Anspruch 1, wobei das zweite Datenmuster einen ersten Teil, der die Substitution identifiziert, und einen zweiten Teil, der das substituierte Datenmuster identifiziert, enthält.

6. Verfahren nach Anspruch 1, wobei der Schritt des Bestimmens das Bestimmen umfaßt, ob die codierten Daten das erste Datenmuster und/oder einen ersten Teil des zweiten Datenmusters enthalten.

7. Verfahren nach Anspruch 1, wobei der Schritt des Substituierens durch das zweite Datenmuster das Substituieren durch einen ersten Teil, der substituierte Daten anzeigt, und einen zweiten Teil, der das durch die Substitution entfernte Datenmuster identifiziert, umfaßt.

8. Verfahren nach Anspruch 7, wobei der Schritt des Bestimmens das Bestimmen umfaßt, ob die codierten Daten das erste Datenmuster und/oder das zweite Datenmuster enthalten.

9. Verfahren nach Anspruch 8, ferner mit dem folgenden Schritt:
Übertragen der codierten Daten in verdrahteter Kommunikation.

10. Verfahren nach Anspruch 5, wobei der Datentyp durch Datenmuster identifiziert wird.

11. Verfahren nach Anspruch 7, wobei der Datentyp durch Datenmuster identifiziert wird.

12. Verfahren nach Anspruch 1, wobei der Schritt des Bestimmens das Bestimmen umfaßt, ob die codierten Daten das erste Datenmuster und/oder einen Teil des zweiten Datenmusters enthalten.

13. Verfahren nach Anspruch 1, wobei der Schritt des Bestimmens das Bestimmen umfaßt, ob die codierten Daten das erste Datenmuster und/oder das zweite Datenmuster enthalten, und wobei der Schritt des Bestimmens das Bestimmen umfaßt, ob die codierten Daten eine Variante des ersten Datenmusters und/oder des zweiten Datenmusters enthalten, wobei für eine vorbestimmte Variante entweder des ersten Datenmusters oder des zweiten Datenmusters das zweite Datenmuster substituiert wird.

14. Verfahren nach Anspruch 5, wobei der Schritt des Bestimmens das Bestimmen umfaßt, ob die codierten Daten das erste Datenmuster und/oder das zweite Datenmuster enthalten.

15. Verfahren nach Anspruch 12, wobei der Datentyp durch Datenmuster identifiziert wird.

16. Verfahren nach Anspruch 13, wobei der Datentyp durch Datenmuster identifiziert wird.

17. Verfahren nach Anspruch 1, ferner mit dem folgenden Schritt:
Hinzufügen eines dritten Datenmusters zu einem Anfangsteil der codierten Daten, um den Anfang der codierten Daten anzuzeigen.

18. Verfahren nach Anspruch 17, wobei das dritte Datenmuster mit dem ersten Datenmuster identisch ist.

19. Verfahren nach Anspruch 17, ferner mit dem folgenden Schritt:
Übertragen der codierten Daten von einer drahtlosen Kommunikationseinrichtung.

20. Verfahren nach Anspruch 7, ferner mit dem folgenden Schritt:
Hinzufügen eines dritten Datenmusters zu einem Anfangsteil der codierten Daten, um den Anfang der codierten Daten anzuzeigen.

21. Verfahren nach Anspruch 13, ferner mit dem folgenden Schritt:
Hinzufügen eines dritten Datenmusters zu einem Anfangsteil der codierten Daten, um den Anfang der codierten Daten anzuzeigen.

22. Verfahren nach Anspruch 20, ferner mit dem folgenden Schritt:
Übertragen der codierten Daten in drahtloser Kommunikation.

23. Verfahren nach Anspruch 21, ferner mit dem folgenden Schritt:
Übertragen der codierten Daten in drahtloser Kommunikation.

24. Verfahren nach Anspruch 17, ferner mit dem folgenden Schritt:
Übertragen eines einzigen mit variabler Länge codierten Datensignals, einschließlich des ersten Datenmusters an einem Endteil davon und des dritten Datenmusters an einem Anfangsteil davon, von einer drahtlosen Kommunikationseinrichtung.

25. Verfahren nach Anspruch 24, wobei beim Codieren zur nachfolgenden Übertragung mehrerer codierter Datensignale das dritte Datenmuster zu dem Anfang eines ersten zu übertragenden codierten Datensignals hinzugefügt, das erste Datenmuster zu dem Ende eines letzten zu übertragenden codierten Datensignals hinzugefügt und das dritte Datenmuster zwischen aufeinanderfolgenden zu übertragenden codierten Datensignalen hinzugefügt wird.

26. Verfahren nach Anspruch 1, ferner mit dem folgenden Schritt:
Codieren von Eingangsdaten durch Vorwärtsfehlerkorrektur vor dem Schritt des Bestimmens.

27. Verfahren nach Anspruch 1, wobei das erste Datenmuster in Speicher vorgespeichert wird.

28. Verfahren nach Anspruch 6, wobei das erste und das zweite Datenmuster in Speicher vorgespeichert werden.

29. Verfahren nach Anspruch 13, wobei das erste und das zweite Datenmuster und Varianten des ersten und des zweiten Datenmusters in Speicher vorgespeichert werden.

30. Verfahren nach Anspruch 29, ferner mit dem folgenden Schritt:
Hinzufügen eines dritten Datenmusters zu einem Anfangsteil der codierten Daten, um den Anfang der codierten Daten anzuzeigen.

31. Verfahren nach Anspruch 1, wobei das Verfahren für CDMA-Anwendungen benutzt wird.

32. Verfahren nach Anspruch 1, wobei das Verfahren für TDMA-Anwendungen benutzt wird.

33. Verfahren nach Anspruch 17, wobei das erste Datenmuster ein Teilmuster des dritten Datenmusters ist.

34. Verfahren nach Anspruch 17, wobei das dritte Datenmuster einen ersten Teil, der den Anfang der mit variabler Länge codierten Daten anzeigt, und einen zweiten Synchronisationsteil enthält.

35. Verfahren nach Anspruch 20, wobei das erste Datenmuster ein Teilmuster des dritten Datenmusters ist.

36. Verfahren nach Anspruch 20, wobei das dritte Datenmuster einen ersten Teil, der den Anfang der codierten Daten anzeigt, und einen zweiten Synchronisationsteil enthält.

37. Verfahren nach Anspruch 21, wobei das erste Datenmuster ein Teilmuster des dritten Datenmusters ist.

38. Verfahren nach Anspruch 21, wobei das dritte Datenmuster einen ersten Teil, der den Anfang der codierten Daten anzeigt, und einen zweiten Synchronisationsteil enthält.

39. Verfahren nach Anspruch 29, wobei das erste Datenmuster ein Teilmuster des dritten Datenmusters ist.

40. Verfahren nach Anspruch 29, wobei das dritte Datenmuster einen ersten Teil, der den Anfang der codierten Daten anzeigt, und einen zweiten Synchronisationsteil enthält.

41. Vorrichtung zum Codieren von Daten, umfassend:
einen Konverter (**220**), der so ausgelegt ist, daß er erkennt, ob ein erstes Datenmuster in den codierten Daten vorhanden ist, und der so ausgelegt ist, daß er ein erkanntes erstes Datenmuster durch ein anderes Datenmuster substituiert;
eine Hinzufügevorrichtung (**230**), die so ausgelegt ist, daß sie das erste Datenmuster zu einem Endteil der codierten Daten hinzufügt, um das Ende der codierten Daten anzuzeigen;
dadurch gekennzeichnet, daß
der Konverter ferner so ausgelegt ist, daß er erkennt, ob eine Variante des ersten Datenmusters vorhanden ist, und ferner so ausgelegt ist, daß er eine erkannte Variante des ersten Datenmusters durch ein anderes Datenmuster substituiert.

42. Vorrichtung nach Anspruch 41, ferner umfassend:
einen Codierer (**204**) zum Codieren von Eingangsdaten, um die in den Konverter eingegebenen codierten Daten zu erzeugen.

43. Vorrichtung nach Anspruch 41, ferner umfassend:
eine Übertragungsvorrichtung (**250**) zum Übertragen der codierten Daten.

44. Vorrichtung nach Anspruch 41, ferner umfassend:
eine Übertragungsvorrichtung zum Übertragen der codierten Daten in drahtloser Kommunikation.

45. Vorrichtung nach Anspruch 41, wobei der Konverter so ausgelegt ist, daß er erkennt, ob eines von mehreren Datenmustern in den codierten Daten vorhanden ist oder nicht, und so ausgelegt ist, daß er ein erkanntes Datenmuster abhängig von dem erkannten Datenmuster durch ein anderes von mehreren Datenmustern substituiert.

46. Vorrichtung nach Anspruch 45, wobei die mehreren anderen Datenmuster einen ersten Teil und einen zweiten Teil enthalten.

47. Vorrichtung nach Anspruch 46, wobei der erste Teil, der für jedes der mehreren anderen Datenmuster gleich ist, substituierte Daten angibt und der zweite Teil den durch die Substitution entfernten Datentyp angibt.

48. Vorrichtung nach Anspruch 41, wobei die Hinzufügevorrichtung ferner so ausgelegt ist, daß sie ein drittes Datenmuster zu einem Anfangsteil der codierten Daten hinzufügt, um den Anfang der mit variabler Länge codierten Daten anzuzeigen.

49. Vorrichtung nach Anspruch 48, ferner umfassend: einen Codierer zum Codieren von Eingangsdaten, um die in den Konverter eingegebenen codierten Daten zu erzeugen.

50. Vorrichtung nach Anspruch 48, ferner umfassend: eine Übertragungsvorrichtung zum Übertragen der codierten Daten.

51. Vorrichtung nach Anspruch 48, wobei das erste Datenmuster ein Teilmuster des dritten Datenmusters ist.

52. Vorrichtung nach Anspruch 48, wobei das dritte Datenmuster einen ersten Teil, der den Anfang der codierten Daten anzeigt, und einen zweiten Synchronisationsteil enthält.

Es folgen 6 Blatt Zeichnungen

Anhängende Zeichnungen

FIG. 1
(Stand der Technik)

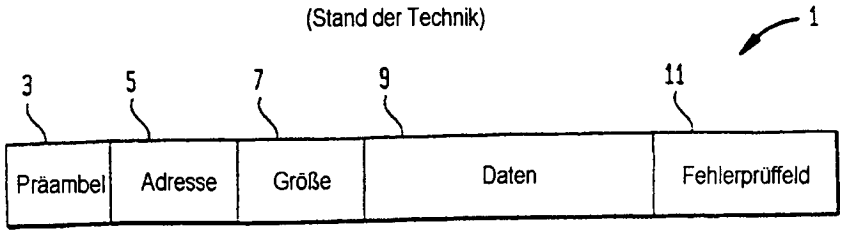


FIG. 2A

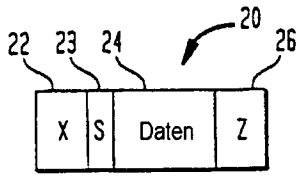


FIG. 2B

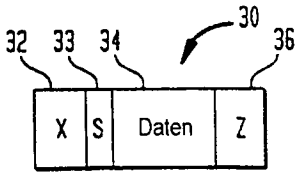


FIG. 2C

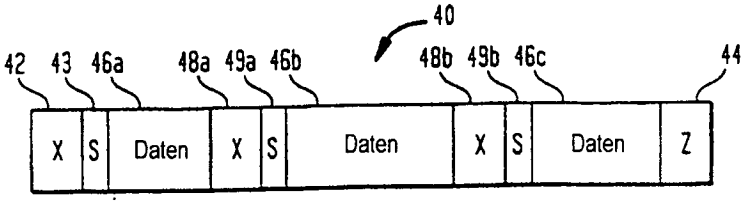


FIG. 3

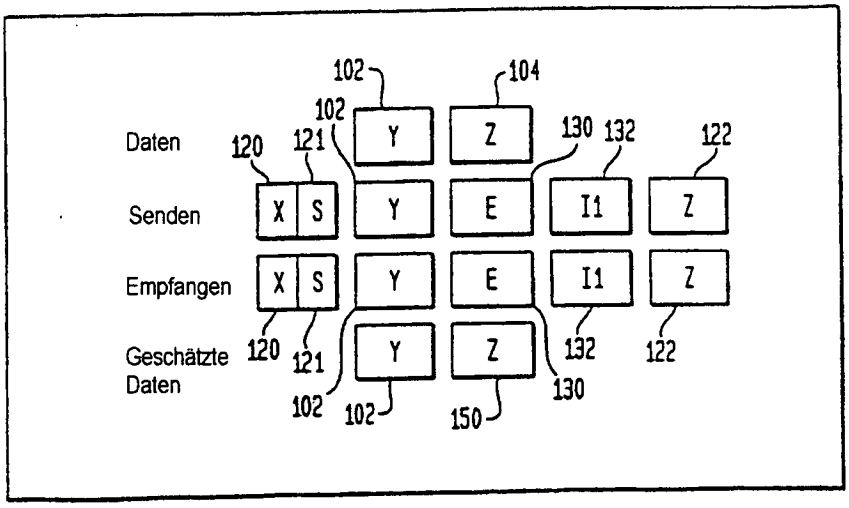


FIG. 4

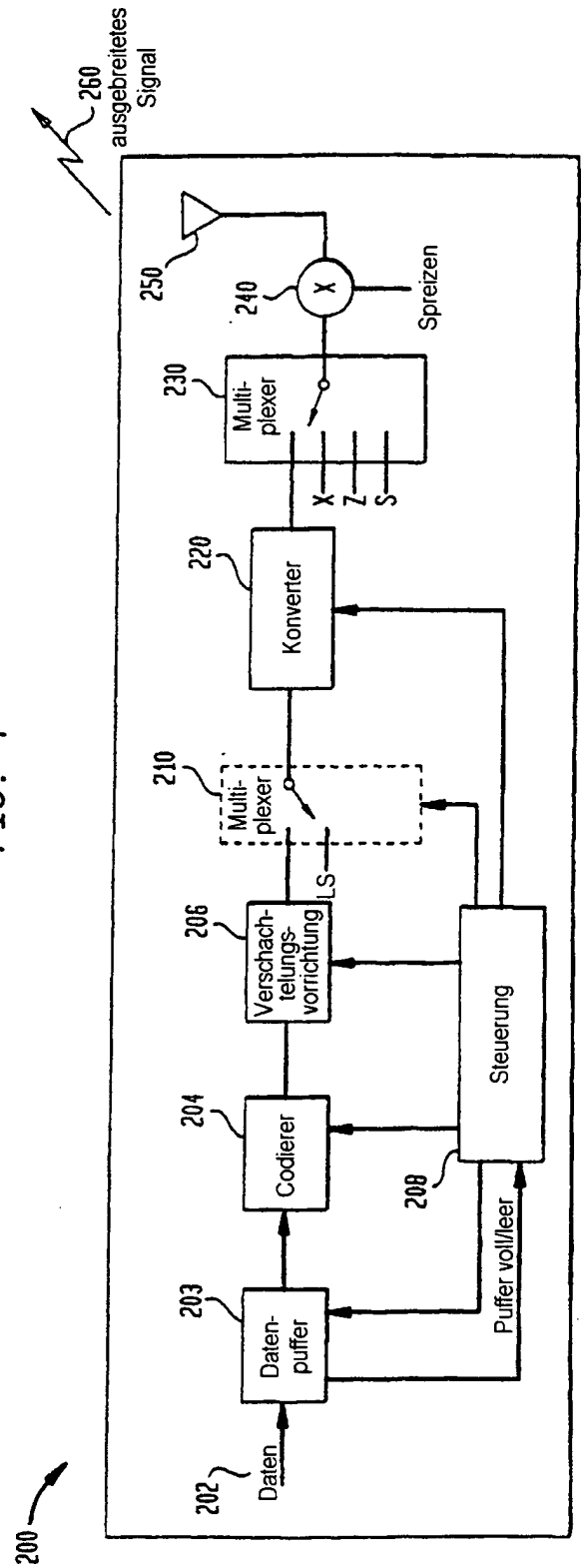

260
ausgebreitetes Signal
250
240
X
Spreizen
230
Multi-plexer
X
Z
S
220
Konverter
210
Multi-plexer
LS
206
Verschach-telungs-vorrichtung
204
Codierer
208
Steuerung
203
Daten-puffer
Puffer voll/leer
202
Daten
200

FIG. 5

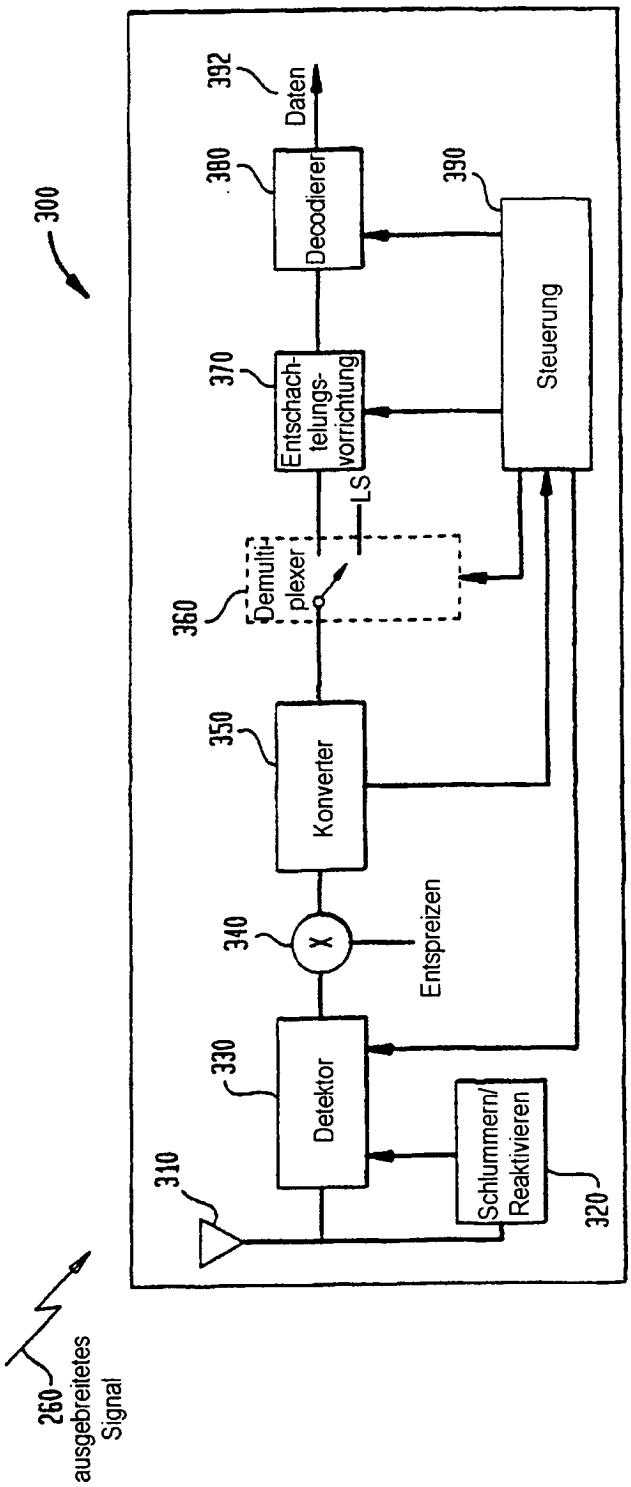

300
260
ausgebreitetes Signal
310
Detektor
330
Schlummern/ Reaktivieren
320
340
X
Entspreizen
Konverter
350
360
Demulti-plexer
LS
Entschach-telungs-vorrichtung
370
Decodierer
380
Steuerung
390
Daten
392

FIG. 6

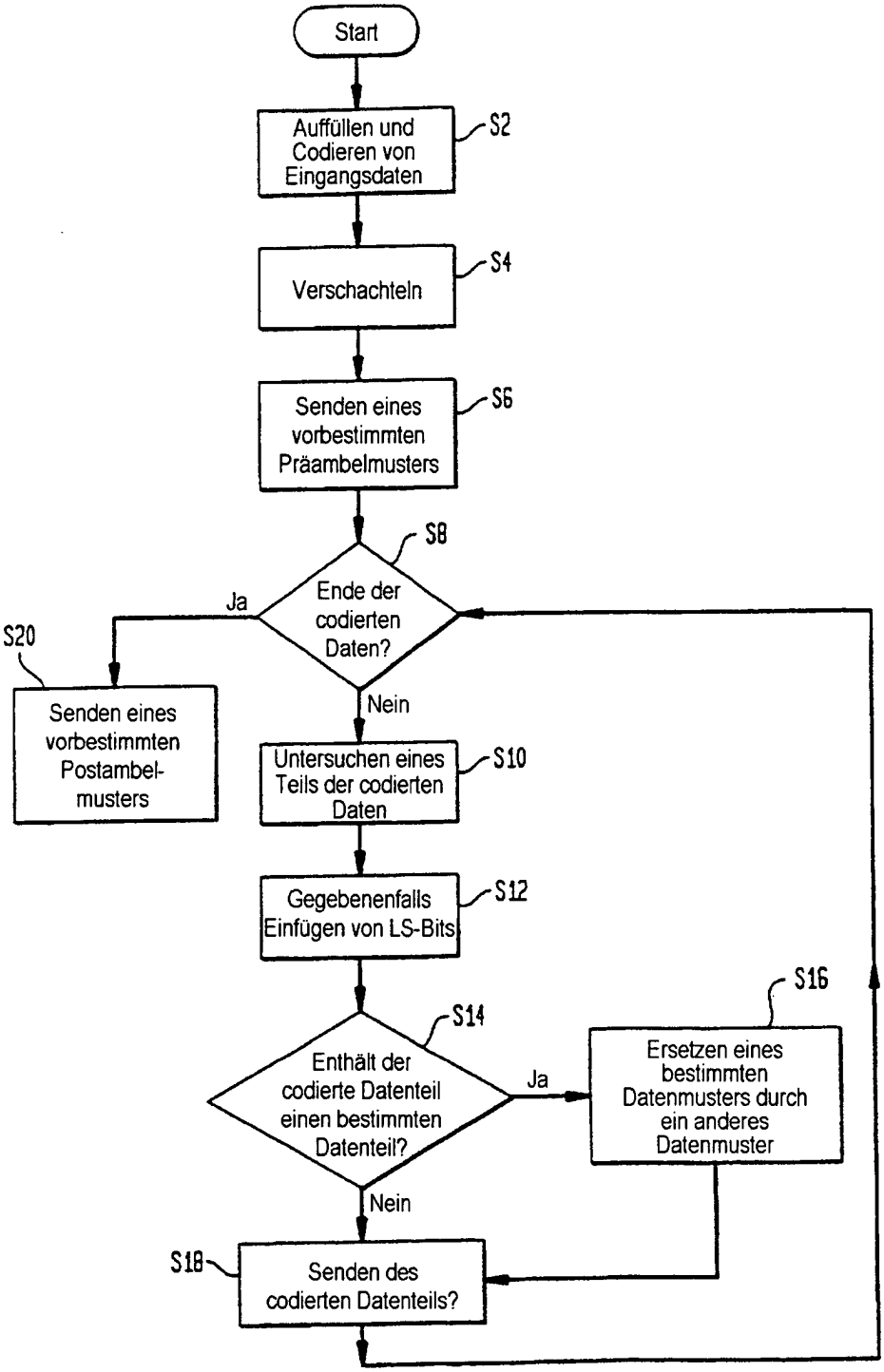

Start
Auffüllen und Codieren von Eingangsdaten
S2
Verschachteln
S4
Senden eines vorbestimmten Präambelmusters
S6
S8
Ende der codierten Daten?
Ja
Nein
S20
Senden eines vorbestimmten Postambel-musters
Untersuchen eines Teils der codierten Daten
S10
Gegebenenfalls Einfügen von LS-Bits
S12
S14
Enthält der codierte Datenteil einen bestimmten Datenteil?
Ja
S16
Ersetzen eines bestimmten Datenmusters durch ein anderes Datenmuster
Nein
S18
Senden des codierten Datenteils?

FIG. 7

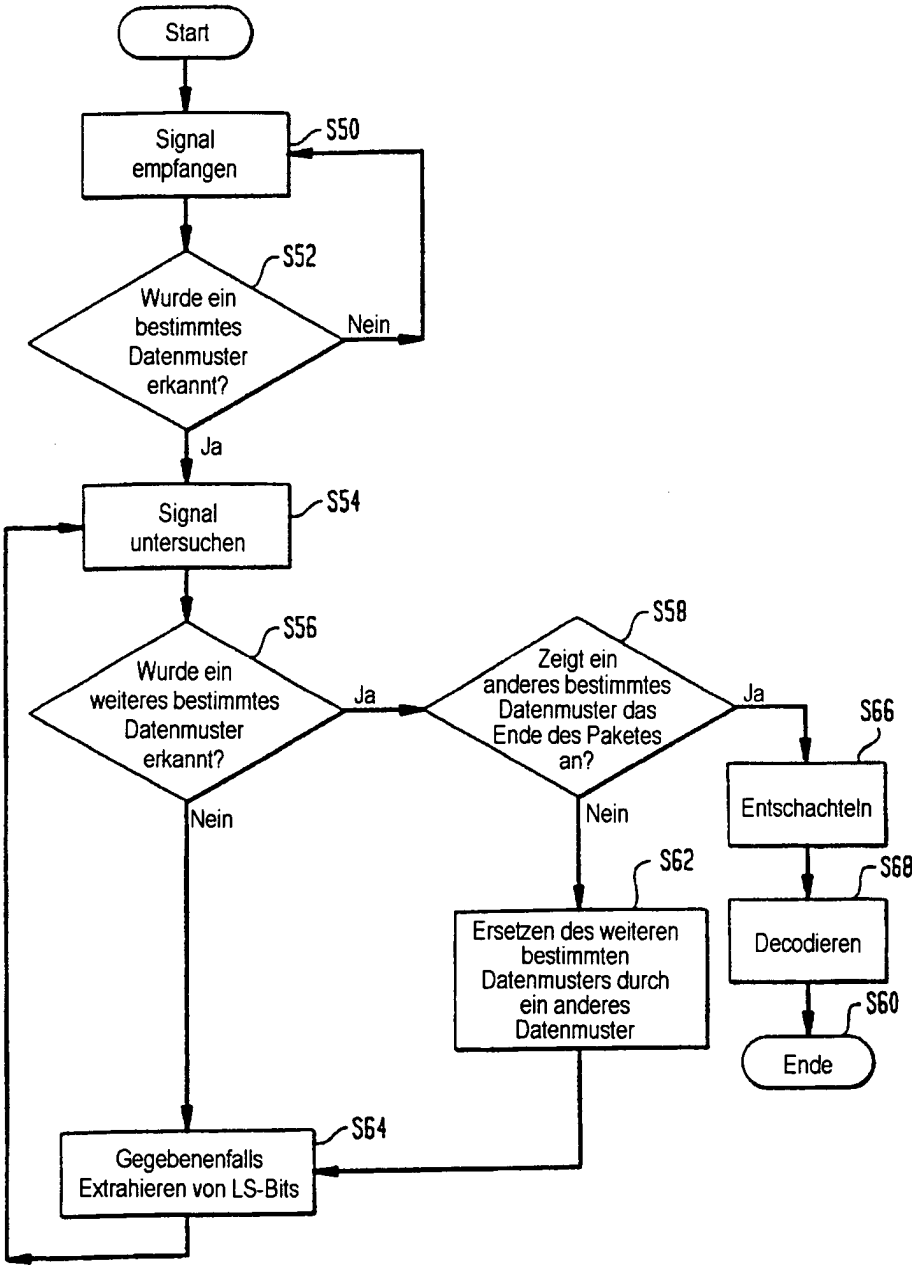

Start
Signal empfangen
S50
Wurde ein bestimmtes Datenmuster erkannt?
S52
Nein
Ja
Signal untersuchen
S54
Wurde ein weiteres bestimmtes Datenmuster erkannt?
S56
Ja
Nein
Zeigt ein anderes bestimmtes Datenmuster das Ende des Paketes an?
S58
Ja
Nein
Entschachteln
S66
Decodieren
S68
Ende
S60
Ersetzen des weiteren bestimmten Datenmusters durch ein anderes Datenmuster
S62
Gegebenenfalls Extrahieren von LS-Bits
S64

FIG. 8

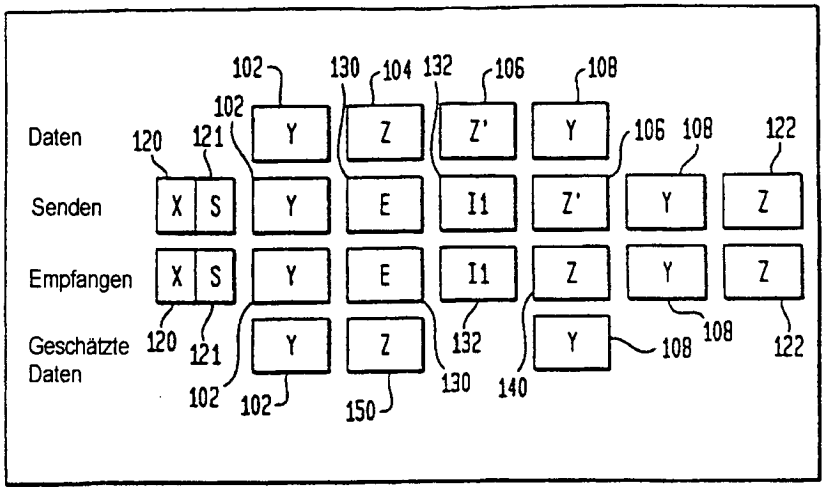
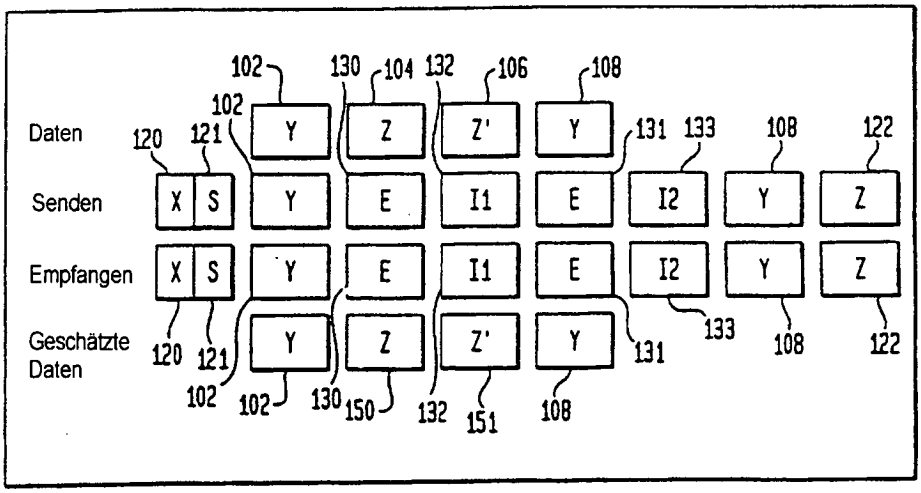

Daten
Senden
Empfangen
Geschätzte Daten
X
S
Y
Z
Z'
E
I1
102
104
106
108
120
121
122
130
132
140
150

FIG. 9


Daten
Senden
Empfangen
Geschätzte Daten
X
S
Y
Z
Z'
E
I1
I2
102
104
106
108
120
121
122
130
131
132
133
150
151