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

(10) **DE 693 32 993 T2** 2004.05.19

(12) # Übersetzung der europäischen Patentschrift

(97) **EP 1 061 505 B1**
(21) Deutsches Aktenzeichen: **693 32 993.9**
(96) Europäisches Aktenzeichen: **00 116 192.6**
(96) Europäischer Anmeldetag: **18.03.1993**
(97) Erstveröffentlichung durch das EPA: **20.12.2000**
(97) Veröffentlichungstag der Patenterteilung beim EPA: **14.05.2003**
(47) Veröffentlichungstag im Patentblatt: **19.05.2004**

(51) Int Cl.[7]: **G10L 19/12**

(30) Unionspriorität:
9142292 **18.03.1992** **JP**
9225992 **18.03.1992** **JP**

(84) Benannte Vertragsstaaten:
DE, FR, GB

(73) Patentinhaber:
Sony Corp., Tokio/Tokyo, JP

(74) Vertreter:
Mitscherlich & Partner, Patent- und Rechtsanwälte, 80331 München

(72) Erfinder:
Nishiguchi, Masayuki, Shinagawa-ku, Tokyo, JP; Matsumoto, Jun, Shinagawa-ku, Tokyo, JP; Ono, Shinobu, Shinagawa-ku, Tokyo, JP

(54) Bezeichnung: **Hocheffizientes Kodierverfahren**

Beschreibung

TECHNISCHES GEBIET

[0001] Diese Erfindung betrifft ein hocheffizientes Codierungsverfahren zum Codieren von Daten auf der Frequenzachse als ein M-dimensionaler Vektor durch Bilden einer Interblockdifferenz von durch Teilen von Eingangsaudiosignalen wie beispielsweise Stimmen- bzw. Sprachsignale und/oder akustische Signale auf der blockweisen Basis erzeugten Daten und Umwandeln der Audiosignale in Signale auf der Frequenzachse.

HINTERGRUNDGEBIET

[0002] Es sind eine Anzahl Codierungsverfahren bekannt, bei denen eine Signalkompression durch Verwendung statistischer Charakteristiken von Audiosignalen, die Stimmen- bzw. Sprachsignale und/oder akustische Signale enthalten, in der Zeitdomäne und in der Frequenzdomäne und Charakteristiken des menschlichen Gehörsinns ausgeführt wird. Diese Codierungsverfahren werden grob in eine Codierung in der Zeitdomäne, eine Codierung in der Frequenzdomäne und eine Analyse-Synthese-Codierung eingeteilt.
[0003] Als ein Beispiel einer hocheffizienten Codierung von Stimmen- bzw. Sprachsignalen ist es, wenn verschiedene Informationsdaten wie beispielsweise eine Spektrumamplitude oder Parameter derselben, beispielsweise LSP-Parameter, α-Parameter oder k-Parameter, quantisiert werden, bei einer Partiellautokorrelations-Analyse-Synthese-Codierung (PARCOR-Analyse-Synthese-Codierung), Multibanderregungscodierung (MBE-Codierung), Einzelbanderregungscodierung (SBE-Codierung), Oberschwingungscodierung, Seitenbandcodierung (SBC), Linearvorhersagecodierung (LPC), diskreten Cosinustransformation (DCT), modifizierten DCT (MDCT) oder schnellen Fouriertransformation (FFT), üblich, eine skalare Quantisierung auszuführen.
[0004] Indessen können bei dem Stimmen- bzw. Sprach-Analyse-Synthese-System wie dem PARCOR-Verfahren, da die Zeitsteuerung des Umschaltens der Erregungsquelle auf der Zeitachse auf der Block-um-Block-Basis (Rahmen-um-Rahmen-Basis) bzw. blockweisen Basis (rahmenweisen Basis) ist, stimmhafte und nicht stimmhafte Töne nicht gemeinsam im gleichen Rahmen existieren. Dies hat zur Folge, dass es unmöglich ist, Laute bzw. Stimmen hoher Qualität zu erzeugen.
[0005] Jedoch bei der MBE-Codierung wird das Band für Laute bzw. Stimmen in einem einzelnen Block (Rahmen) in mehrere Bänder geteilt und für jedes der Bänder eine Stimmhaft/Stimmlos-Entscheidung getroffen. Infolgedessen können Verbesserungen der Schall- bzw. Klang- bzw. Tonqualität beobachtet werden. Jedoch ist die MBE-Codierung hinsichtlich der Bitrate unvorteilhaft, da für jedes Band erhaltene Stimmhaft/Stimmlos-Entscheidungsdaten separat übertragen werden müssen.
[0006] Auch ist eine skalare Quantisierung wegen des erhöhten Quantisierungsrauschens schwierig auszuführen, wenn versucht wird, für eine weitere Erhöhung der Quantisierungseffizienz die Bitrate auf beispielsweise drei bis vier kbit/s abzusenken.
[0007] Es kann in Erwägung gezogen werden, eine Vektorquantisierung anzunehmen. Jedoch wird mit der Zahl b von Bits eines Ausgangssignals (Index) der Vektorquantisierung die Größe eines Codebuchs eines Vektorquantisierers proportional zu 2^b erhöht, und das Operationsvolumen für eine Codebuchsuche wird ebenfalls proportional zu 2^b erhöht. Da jedoch eine extrem kleine Zahl b von Bits eines Ausgangssignals das Quantisierungsrauschen erhöht, ist es wünschenswert, die Größe des Codebuchs oder die Operationsmenge für eine Codebuchsuche zu reduzieren, während ein gewisser größerer Wert der Bitzahl b beibehalten wird. Nebenbei bemerkt kann die Codierungseffizienz nicht ausreichend erhöht werden, wenn die in jene auf der Frequenzachse umgewandelten Daten durch eine Vektorquantisierung direkt verarbeitet werden. Infolgedessen ist eine Technik für eine weitere Erhöhung des Kompressionsverhältnisses erforderlich.
[0008] Im Hinblick auf den oben beschriebenen Stand der Technik ist es eine Aufgabe der vorliegenden Erfindung, ein hocheffizientes Codierungsverfahren bereitzustellen, wodurch die für jedes Band erzeugten Entscheidungsdaten für stimmhafte/stimmlose Töne mit einer reduzierten Zahl von Bits ohne Verschlechterung der Tonqualität übertragen werden können.
[0009] Es ist eine andere Aufgabe der vorliegenden Erfindung, ein hocheffizientes Codierungsverfahren bereitzustellen, wodurch die Größe des Codebuchs für den Vektorquantisierer oder das Operationsvolumen für Codebuchsuche ohne Erniedrigung der Zahl Ausgangsbits einer Vektorquantisierung verkleinert werden kann, und wodurch das Kompressionsverhältnis zum Zeitpunkt der Vektorquantisierung weiter erhöht werden kann.

OFFENBARUNG DER ERFINDUNG

[0010] Gemäß der vorliegenden Erfindung ist ein hocheffizientes Codierungsverfahren bereitgestellt, das die Schritte aufweist: Finden von Interblockdifferenzdaten als einen M-dimensionalen Vektor, wobei M eine ganze Zahl größer als eins ist, durch Nehmen bzw. Bilden einer Interblockdifferenz von durch Teilen eines Eingangsaudiosignals in Blöcke und Umwandeln der resultierenden Blocksignale in Signale auf einer Frequen-

zachse, und Verarbeiten der Interblockdifferenzdaten des M-dimensionalen Vektors durch Vektorquantisierung.

KURZE BESCHREIBUNG DER ZEICHNUNGEN

[0011] **Fig.** 1 ist ein funktionelles Blockschaltbild, das eine schematische Anordnung einer Analyseseite oder Codiererseite einer Synthese-Analyse-Codierungseinrichtung für Stimmen- bzw. Sprachsignale als ein spezifisches Beispiel einer Einrichtung ist, auf die ein hocheffizientes Codierungsverfahren der vorliegenden Erfindung angewendet ist.
[0012] **Fig.** 2 ist ein Diagramm zur Erläuterung einer Fensterverarbeitung.
[0013] **Fig.** 3 ist ein Diagramm zur Erläuterung einer Relation zwischen der Fensterverarbeitung und einer Fensterfunktion.
[0014] **Fig.** 4 ist ein Diagramm, das Zeitachsendaten als ein Objekt einer Orthogonaltransformationsverarbeitung (FFT-Verarbeitung) zeigt.
[0015] **Fig.** 5 ist ein Diagramm, das ein Leistungsspektrum von Spektrumdaten, eine Spektrumenveloppe und Erregungssignale auf der Frequenzachse zeigt.
[0016] **Fig.** 6 ist ein funktionelles Blockschaltbild, das eine schematische Anordnung einer Syntheseseite oder Decodiererseite der Synthese-Analyse-Codierungseinrichtung für Sprachsignale als ein konkretes Beispiel einer Einrichtung ist, auf die das hocheffiziente Codierungsverfahren der vorliegenden Erfindung angewendet ist.
[0017] **Fig.** 7 ist ein Diagramm zur Erläuterung einer Stimmlostonsynthese zum Zeitpunkt der Synthese von Sprachsignalen.
[0018] **Fig.** 8 ist ein Wellenformdiagramm zur Erläuterung eines herkömmlichen Tonhöhenextraktionsverfahrens.
[0019] **Fig.** 9 ist ein funktionelles Blockschaltbild zur Erläuterung eines ersten Beispiels des beim hocheffizienten Codierungsverfahren gemäß der vorliegenden Erfindung verwendeten Tonhöhenextraktionsverfahrens.
[0020] **Fig.** 10 ist ein Flussdiagramm zur Erläuterung einer Bewegung des ersten Beispiels des Tonhöhenextraktionsverfahrens.
[0021] **Fig.** 11 ist ein Wellenformdiagramm zur Erläuterung des ersten Beispiels des Tonhöhenextraktionsverfahrens.
[0022] **Fig.** 12 ist ein funktionelles Blockschaltbild, das eine schematische Anordnung eines konkreten Beispiels zeigt, auf das ein zweites Beispiel des bei dem hocheffizienten Codierungsverfahren der vorliegenden Erfindung verwendeten Tonhöhenextraktionsverfahrens angewendet ist.
[0023] **Fig.** 13 ist ein Wellenformdiagramm zur Erläuterung einer Verarbeitung einer Eingangssprachsignalwellenform des zweiten Beispiels des Tonhöhenextraktionsverfahrens.
[0024] **Fig.** 14 ist ein Flussdiagramm zur Erläuterung einer Bewegung der Tonhöhenextraktion im zweiten Beispiel des Tonhöhenextraktionsverfahrens.
[0025] **Fig.** 15 ist ein funktionelles Blockschaltbild, das eine schematische Anordnung eines konkreten Beispiels zeigt, auf das ein drittes Beispiel des Tonhöhenextraktionsverfahrens angewendet ist.
[0026] **Fig.** 16 ist ein Wellenformdiagramm zur Erläuterung einer herkömmlichen Sprachcodierung.
[0027] **Fig.** 17 ist ein Flussdiagramm zur Erläuterung einer Bewegung einer Codierung eines Beispiels eines bei dem hocheffizienten Codierungsverfahren der vorliegenden Erfindung angewendeten Sprachcodierungsverfahrens.
[0028] **Fig.** 18 ist ein Wellenformdiagramm zur Erläuterung einer Codierung eines Beispiels des Sprachcodierungsverfahrens.
[0029] **Fig.** 19 ist ein Flussdiagramm zur Erläuterung wesentlicher Abschnitte einer Ausführungsform des hocheffizienten Codierungsverfahrens der vorliegenden Erfindung.
[0030] **Fig.** 20 ist ein Diagramm zur Erläuterung einer Feststellung eines Grenzpunktes einer Stimmhaft(V)/Stimmlos(UV)-Tonabgrenzung eines Bandes.
[0031] **Fig.** 21 ist ein Blockschaltbild, das eine schematische Anordnung zur Erläuterung einer Umwandlung der Zahl von Daten zeigt.
[0032] **Fig.** 22 ist ein Wellenformdiagramm zur Erläuterung eines Beispiels einer Umwandlung der Zahl von Daten.
[0033] **Fig.** 23 ist ein Diagramm, das ein Beispiel einer Wellenform für eine expandierte Zahl von Daten vor einer FFT zeigt.
[0034] **Fig.** 24 ist ein Diagramm, das ein Vergleichsbeispiel der Wellenform für die expandierte Zahl von Daten vor der FFT zeigt.
[0035] **Fig.** 25 ist ein Diagramm zur Erläuterung einer Wellenform nach der FFT und einer Überabtastoperation.
[0036] **Fig.** 26 ist ein Diagramm zur Erläuterung einer Filterungsoperation bei der Wellenform nach der FFT.

[0037] **Fig.** 27 ist ein Diagramm, das eine Wellenform nach einer IFFT zeigt.
[0038] **Fig.** 28 ist ein Diagramm, das ein Beispiel einer Umwandlung der Zahl von Abtastwerten durch Überabtastung zeigt.
[0039] **Fig.** 29 ist ein Diagramm zur Erläuterung einer linearen Kompensations- und Beschränkungsverarbeitung zeigt.
[0040] **Fig.** 30 ist ein Blockschaltbild, das eine schematische Anordnung eines Codierers zeigt, auf den das hocheffiziente Codierungsverfahren der vorliegenden Erfindung angewendet ist.
[0041] **Fig.** 31 bis 36 sind Diagramme zur Erläuterung einer Bewegung einer Vektorquantisierung einer hierarchischen Struktur.
[0042] **Fig.** 37 ist ein Blockschaltbild, das eine schematische Anordnung eines Codierers zeigt, auf den ein anderes Beispiel des hocheffizienten Codierungsverfahrens angewendet ist.
[0043] **Fig.** 38 ist ein Blockschaltbild, das eine schematische Anordnung eines Codierers zeigt, auf den ein noch anderes Beispiel des hocheffizienten Codierungsverfahrens angewendet ist.
[0044] **Fig.** 39 ist ein Blockschaltbild, das eine schematische Anordnung eines Codierers zeigt, auf den ein hocheffizientes Codierungsverfahren zum Umschalten eines Codebuches einer Vektorquantisierung entsprechend Eingangssignalen angewendet ist.
[0045] **Fig.** 40 ist ein Diagramm zur Erläuterung eines Trainingsverfahrens des Codebuches.
[0046] **Fig.** 41 ist ein Blockschaltbild, das eine schematische Anordnung wesentlicher Abschnitte eines Codierers zur Erläuterung eines anderen Beispiels des hocheffizienten Codierungsverfahrens zum Umschalten des Codebuches zeigt.
[0047] **Fig.** 42 ist eine schematische Darstellung zur Erläuterung eines herkömmlichen Vektorquantisierers.
[0048] **Fig.** 43 ist ein Flussdiagramm zur Erläuterung eines LBG-Algorithmus.
[0049] **Fig.** 44 ist eine schematische Darstellung zur Erläuterung eines ersten Beispiels eines Vektorquantisierungsverfahrens.
[0050] **Fig.** 45 ist ein Diagramm zur Erläuterung von Kommunikationsfehlern in einem generellen Kommunikationssystem, das zur Erläuterung eines zweiten Beispiels des Vektorquantisierungsverfahrens verwendet ist.
[0051] **Fig.** 46 ist ein Flussdiagramm zur Erläuterung des zweiten Beispiels des Vektorquantisierungsverfahrens.
[0052] **Fig.** 47 ist eine schematische Darstellung zur Erläuterung eines dritten Beispiels des Vektorquantisierungsverfahrens.
[0053] **Fig.** 48 ist ein funktionelles Blockschaltbild eines konkreten Beispiels, bei dem ein Sprach-Analyse-Synthese-Verfahren auf einen sogenannten Vocoder angewendet ist.
[0054] **Fig.** 49 ist ein Graph zur Erläuterung eines bei dem Sprach-Analyse-Synthese-Verfahren angewendeten Gaußschen Rauschens.

BESTE ART UND WEISE DER AUSFÜHRUNG DER ERFINDUNG

[0055] Unter Bezugnahme auf die Zeichnungen werden bevorzugte Ausführungsformen des hocheffizienten Codierungsverfahrens gemäß der vorliegenden Erfindung erläutert.
[0056] Es ist für das hocheffiziente Codierungsverfahren möglich, ein Codierungsverfahren zu verwenden, das eine Umwandlung von Signalen auf der Block-um-Block-Basis bzw. blockweisen Basis in Signale auf der Frequenzachse, Teilen des Frequenzbandes der resultierenden Signale in mehrere Bänder und voneinander Unterscheiden von stimmhaften (V) und stimmlosen (UV) Tönen für jedes der Bänder wie im Fall des später erläuterten Multibanderregungscodierungsverfahrens (MBE-Verfahren) aufweist.
[0057] Das heißt, bei einem generellen hocheffizienten Codierungsverfahren gemäß der vorliegenden Erfindung wird ein Stimmen- bzw. Sprachsignal in Blöcke geteilt, deren jeder aus einer vorbestimmten Zahl von Abtastpunkten bzw. Abtastwerten, beispielsweise 256 Abtastwerten besteht, und das resultierende Signal auf der blockweisen Basis wird durch eine Orthogonaltransformation wie beispielsweise FFT in Spektrumdaten auf der Frequenzachse umgewandelt. Gleichzeitig wird die Tonstärke bzw. Tonhöhe der Stimme bzw. Sprache in jedem Block extrahiert, und das Spektrum auf der Frequenzachse wird in einem Intervall entsprechend der Tonlage bzw. Tonhöhe in mehrere Bänder geteilt. Dann wird für jedes der geteilten Bänder eine Stimmhaft(V)/Stimmlos(W)-Tonunterscheidung getroffen. Die V/W-Tonunterscheidungsdaten werden codiert und zusammen mit Spektrumamplitudendaten übertragen.
[0058] Ein konkretes Beispiel eines Multibanderregungsvocoders (MBE-Vocoder), der eine Art Synthese-Analyse-Codierer für Sprachsignale (ein sogenannter Vocoder) ist, auf den das hocheffiziente Codierungsverfahren der vorliegenden Erfindung angewendet werden kann, wird nachfolgend unter Bezugnahme auf die Zeichnungen erläutert.
[0059] Der nun zu erläuternde MBE-Vocoder geht aus D. W. Griffin and J. S. Lim, „Multiband Excitation Vocoder", IEEE Trans. Acoustics, Speech and Signal Processing, Vol. 36, Nr. 8, August 1988, Seiten 1223–1235 hervor. Im Gegensatz zu einem herkömmlichen Partiellautokorrelationsvocoder (PARCOR-Vocoder), bei dem

zum Zeitpunkt der Stimmen- bzw. Sprachmodellierung stimmhafte Bereiche und stimmlose Bereiche auf der blockweisen Basis oder auf der Rahmen-um-Rahmen-Basis bzw. rahmenweisen Basis umgeschaltet werden, führt der MBE-Vocoder eine Modellierung unter der Annahme aus, dass ein stimmhafter Bereich und ein stimmloser Bereich in einem gleichzeitigen Bereich auf der Frequenzachse, das heißt im gleichen Block oder Rahmen existiert.

[0060] Die **Fig.** 1 ist ein schematisches Blockschaltbild, das eine Gesamtanordnung einer Ausführungsform des MBE-Vocoders zeigt, auf den die vorliegende Erfindung angewendet ist.

[0061] Was die **Fig.** 1 betrifft, so wird einem Eingangsanschluss **101** ein Sprachsignal zugeführt und dann zu einem Filter **102**, beispielsweise ein Hochpassfilter (HPF), übertragen, um von einer sogenannten Gleichsignalverschiebung (DC-Verschiebung) und wenigstens von Niederfrequenzkomponenten von nicht höher als 200 Hz zur Begrenzung des Frequenzbandes auf beispielsweise 200 bis 3400 Hz befreit zu werden. Ein vom Filter **102** erhaltenes Signal wird einem Tonhöhenextraktionsabschnitt **103** und einem Fensterverarbeitungsabschnitt **104** zugeführt. Der Tonhöhenextraktionsabschnitt **103** teilt Eingangssprachsignaldaten in Blöcke, deren jeder aus einer vorbestimmten Zahl oder N Abtastwerten, beispielsweise 256 Abtastwerte, besteht, und/oder schneidet mittels eines Rechteckfensters aus und führt eine Tonhöhenextraktion für Sprachsignale in jedem Block aus. Diese Blöcke, deren jeder aus 256 Abtastwerten besteht, werden, wie bei A in **Fig.** 5 gezeigt, entlang der Zeitachse in einem Intervall eines L Abtastwerte, beispielsweise 160 Abtastwerte aufweisenden Rahmens bewegt, so dass eine gegenseitige Blocküberlappung bzw. Interblocküberlappung gleich (N – L) Abtastwerte, beispielsweise 96 Abtastwerte beträgt. Der Fensterverarbeitungsabschnitt **104** multipliziert die N Abtastwerte jedes Blocks mit einer vorbestimmten Fensterfunktion, beispielsweise einem Hammingfenster, und die mit Fenstern versehenen Blöcke werden entlang der Zeitachse in einem Intervall von L Abtastwerten pro Rahmen sequentiell bewegt.

[0062] Diese Fensterverarbeitung kann durch die Formel

$$X_W(k, q) = x(q)\, w(kl - q) \qquad (1)$$

ausgedrückt werden, wobei k eine Blockzahl und q einen Zeitindex einer Daten- oder Abtastwertzahl bezeichnet. Die Formel zeigt, dass die q-ten Daten eines Eingangssignals x(q) vor der Verarbeitung mit einer Fensterfunktion des k-ten Blocks w(kl – q) multipliziert wird, um Daten $x_w(k, q)$ zu ergeben. Die Fensterfunktion $w_r(r)$ für ein bei A in **Fig.** 2 gezeigtes rechteckiges Fenster im Tonhöhenextraktionsabschnitt **103** wird

$$\begin{aligned} w_r(r) &= 1 && 0 \le r < N \qquad (2)\\ &= 0 && r < 0,\ N \le r \end{aligned}$$

ausgedrückt.

[0063] Die Fensterfunktion $w_h(r)$ für ein bei B in **Fig.** 2 gezeigtes Hammingfenster beim Fensterverarbeitungsabschnitt **104** ist durch

$$\begin{aligned} w_h(r) &= 0{,}54 - 0{,}46 \cos(2\pi r/(N-1)) \\ &\qquad 0 \le r < N \qquad (3)\\ &= 0 \qquad r < 0,\ N \le r \end{aligned}$$

gegeben.

[0064] Bei Verwendung der Fensterfunktion $w_r(r)$ oder $w_h(r)$ ist eine von Null verschiedene Domäne der Fensterfunktion w(r) (= w(kl – q)) der obigen Formel (1) durch

$$0 \le kL - q < N$$

gegeben, und eine Modifikation von diesem wird durch die folgende Formel

$$kL - N < q \le kL$$

ausgedrückt.

[0065] Deshalb gilt bei kL – N < q ≤ kL, dass die Fensterfunktion $w_r(kL - q) = 1$ wie in **Fig.** 3 gezeigt für das rechtwinkelige Fenster steht. Die obigen Formeln (1) bis (3) zeigen an, dass das eine Länge von N(= 256) Abtastwerten aufweisende Fenster zu einem Zeitpunkt mit einer Rate von L(= 160) Abtastwerten vorbewegt wird. Von Null verschiedene Abtastwertzüge bei jedem Punkt N(0 ≤ r < N) geteilt durch jede der Fensterfunktionen der Formeln (2) und (3) sind durch $x_{wr}(k, r)$ bzw. $x_{wh}(k, r)$ angedeutet.

[0066] Der Fensterverarbeitungsabschnitt **104** addiert 0-Daten für 1792 Abtastwerte zu einem 256-Abtastwertblock-Abtastwertzug $x_{wh}(k, r)$ multipliziert mit dem Hammingfenster der Formel (3), wodurch, wie in **Fig.** 4

gezeigt, 2048 Abtastwerte erzeugt werden. Die Datenfolge von 2048 Abtastwerten auf der Zeitachse werden von einem Orthogonaltransformationsabschnitt **105** mit einer Orthogonaltransformation wie beispielsweise einer schnellen Fouriertransformation verarbeitet.

[0067] Der Tonhöhenextraktionsabschnitt **103** führt eine Tonhöhenextraktion auf der Basis des obigen Einzelblock-N-Abtastwert-Abtastwertzugs $x_{wr}(k, r)$ aus. Obgleich eine Tonhöhenextraktion unter Verwendung einer Periodizität der zeitlichen Wellenform, einer periodischen spektralen Frequenzstruktur oder einer Autokorrelationsfunktion ausgeführt werden kann, ist bei der vorliegenden Ausführungsform das Mitteabschneidewellenform- bzw. Mitteclipwellenform-Autokorrelationsverfahren angenommen. Was den Mitteabschneidepegel bzw. Mitteclippegel in jedem Block betrifft, kann für jeden Block ein einziger Abschneidepegel bzw. Clippegel eingestellt werden. Jedoch wird der Spitzenpegel von Signalen jeder Unterteilung des Blocks (jedes Subblocks) detektiert, und bei einer großen Differenz im Spitzenpegel zwischen den Subblocks wird der Clippegel im Block fortschreitend oder kontinuierlich geändert. Die Spitzenperiode wird auf der Basis des Spitzenabschnitts der Autokorrelationsdaten der zentralen Abschneidewellenform bzw. Clipwellenform bestimmt. Zu diesem Zeitpunkt werden von den zum laufenden Rahmen gehörenden autokorrelierten Daten mehrere Spitzen gefunden, wo die Autokorrelation von 1-Block-N-Abtastwerte-Daten als ein Objekt gefunden werden. Wenn das Maximum einer dieser Spitzen nicht kleiner als eine vorbestimmte Schwelle ist, ist die maximale Spitzenposition die Tonhöhenperiode. Andernfalls wird eine Spitze gefunden, die in einem gewissen Tonhöhenbereich ist, welcher der Relation mit einer Tonhöhe eines von dem laufenden Rahmen verschiedenen Rahmens wie beispielsweise eines vorhergehenden Rahmens oder nachfolgenden Rahmens, zum Beispiel in einem Bereich von ± 20% in Bezug auf die Tonhöhe des vorhergehenden Rahmens, genügt, und die Tonhöhe des laufenden Rahmens wird auf der Basis dieser Spitzenposition bestimmt. Der Tonhöhenextraktionsabschnitt **103** führt eine relativ grobe Tonhöhensuche durch eine offene Schleife aus. Die extrahierten Tonhöhendaten werden einem Feintonhöhensuchabschnitt **106** zugeführt, wo durch eine geschlossene Schleife eine feine Tonhöhensuche ausgeführt wird.

[0068] Vom Tonhöhenextraktionsabschnitt **103** extrahierte ganzzahlig bewertete grobe Tonhöhendaten und Daten auf der Frequenzachse aus dem Orthogonaltransformationsabschnitt **105** werden dem Feintonhöhensuchabschnitt **106** zugeführt. Der Feintonhöhensuchabschnitt **106** erzeugt einen optimalen Feintonhöhendatenwert mit gleitenden Dezimalstellen durch Oszillieren von ± mehreren Abtastwerten mit einer Rate von 0,2 bis 0,5 um den Tonhöhenwert als die Mitte. Als Feinsuchetechnik für die Wahl der Klangfarbe bzw. Tonhöhe wird ein Analyse-durch-Synthese-Verfahren verwendet, so dass das synthetisierte Leistungsspektrum dem Leistungsspektrum des ursprünglichen Tons am nächsten ist.

[0069] Nachfolgend wird die Feintonhöhensuche erläutert. Im MBE-Decodierer ist ein Modell derart angenommen, dass bei ihm S(j) als mit einer Orthogonaltransformation, beispielsweise FFT, verarbeitete Spektrumdaten auf der Frequenzachse durch

$$S(j) = H(j)\,|E(j)| \quad 0 < j < J \tag{4}$$

ausgedrückt sind, wobei J mit $\omega_s/4\pi = f_s/2$ und folglich mit 4 kHz korrespondiert, wenn die Abtastfrequenz fs = ωs/2π gleich 8 kHz ist. In der Formel (4) stellt in dem Fall, dass die Spektrumdaten S(j) auf der Frequenzachse die wie bei A in **Fig.** 5 gezeigte Wellenform aufweisen, H(j) eine bei B in **Fig.** 5 gezeigte Spektrumeinhüllende bzw. Spektrumenveloppe der ursprünglichen Spektrumdaten S(j) dar, während E(j) ein Spektrum eines bei C in **Fig.** 5 gezeigten gleichpegeligen periodischen Erregungssignals darstellt. Das heißt, das FFT-Spektrum S(j) ist in einem Modell als ein Produkt der Spektrumenveloppe H(j) und dem Leistungsspektrum |E(j)| des Erregungssignals angeordnet.

[0070] Das Leistungsspektrum |E(j)| des Erregungssignals wird durch Anordnen der Spektrumwellenform eines Bandes für jedes Band auf der Frequenzachse auf wiederholte Weise gebildet, bei Berücksichtigung der entsprechend der Tonhöhe bestimmten Periodizität (Tonhöhenstruktur) der Wellenform auf der Frequenzachse. Die Einzelbandwellenform kann durch FFT-Verarbeitung der wie in **Fig.** 4 gezeigt aus der 256-Abtastwerte-Hammingfensterfunktion mit dieser hinzugefügten O-Daten von 1792 Abtastpunkten bestehenden Wellenform als Zeitachsensignale und durch Teilen der die Bandbreiten auf der Frequenzachse aufweisenden Impulswellenform entsprechend der obigen Tonhöhe gebildet werden.

[0071] Dann wird für jedes der entsprechend der Tonhöhe geteilten Bänder ein Wert (Amplitude) $|A_m|$, der H(j) darstellt (oder der den Fehler für jedes Band minimiert), gefunden. Wenn ein oberer und unterer Grenzpunkt beispielsweise des m-ten Bandes (Band der m-ten Oberschwingung) auf a_m bzw. b_m eingestellt werden, ist ein Fehler ε_m des m-ten Bandes durch die Formel

$$\varepsilon_m = \sum_{j=a_m}^{b_m} \{|S(j)| - |A_m||E(j)|\}^2 \tag{5}$$

gegeben. Der den Fehler ε_m minimierende Wert von $|A_m|$ ist durch:

$$\frac{\partial \varepsilon_m}{\partial |A_m|} = -2 \sum_{j=a_m}^{b_m} \{|S(j)| - |A_m||E(j)|\}|E(j)|$$

$$|A_m| = \sum_{j=a_m}^{b_m} |S(j)||E(j)| / \sum_{j=a_m}^{b_m} |E(j)|^2 \qquad (6)$$

gegeben. Der Fehler ε_m wird für $|A_m|$ in der obigen Formel (6) minimiert. Eine solche Amplitude $|A_m|$ wird für jedes Band gefunden, und es wird der Fehler ε_m für jedes Band, wie er durch die Formel (5) unter Verwendung jeder Amplitude $|A_m|$ definiert ist, gefunden. Die Summe $\Sigma\varepsilon_m$ aller Bänder wird aus den Fehlern ε_m je Band gefunden. Die Summe $\Sigma\varepsilon_m$ aller Bänder wird für mehrere geringfügig verschiedene Tonhöhen gefunden, und es wird eine Tonhöhe gefunden, welche die Summe $\Sigma\varepsilon_m$ der Fehler minimiert.

[0072] Es werden mehrere Tonhöhen oberhalb und unterhalb der durch den Tonhöhenextraktionsabschnitt **103** in einem Intervall von beispielsweise 0,25 gefundenen groben Tonhöhe erzeugt. Dann wird die Summe $\Sigma\varepsilon_m$ der Fehler für jede der geringfügig verschiedenen Tonhöhen gefunden. Wenn die Tonhöhe bestimmt ist, wird die Bandbreite bestimmt. Unter Verwendung des Leistungsspektrums $|S(j)|$ der Daten auf der Frequenzachse und des Erregungssignalsspektrums $|E(j)|$ wird der Fehler ε_m der Formel (5) aus der Formel (6) gefunden, um die Summe $\Sigma\varepsilon_m$ aller Bänder zu finden. Die Summe $\Sigma\varepsilon_m$ wird für jede Tonhöhe gefunden, und eine Tonhöhe, die mit der minimalen Summe der Fehler korrespondiert, wird als eine optimale Tonhöhe bestimmt. Infolgedessen wird in der Feintonhöhensucheinheit **106** die feinste Tonhöhe (beispielsweise als 0,25-Intervall-Tonhöhe) gefunden, um die mit der optimalen Tonhöhe korrespondierende Amplitude $|A_m|$ zu bestimmen.

[0073] Bei der obigen Erläuterung der Feintonhöhensuche ist der Einfachheit halber angenommen, dass alle Bänder stimmhafter Ton sind. Da jedoch im MBE-Vocoder das Modell angenommen ist, bei welchem auf der gleichlaufenden Frequenzachse ein stimmloser Bereich vorhanden ist, ist es notwendig, für jedes Band eine Unterscheidung zwischen dem stimmhaften Ton und dem stimmlosen Ton zu treffen.

[0074] Daten der optimalen Tonhöhe und Amplitude $|A_m|$ werden vom Feintonhöhensuchabschnitt 106 einem Stimmhaft/Stimmlos-Unterscheidungsabschnitt **107** zugeführt, bei welchen eine Stimmhaft/Stimmlos-Unterscheidung für jedes Band ausgeführt wird. Für eine solche Unterscheidung wird ein Rausch-Signal-Verhältnis (NSR) verwendet. Das heißt, NSR für das m-te Band ist durch die Formel (7)

$$NSR = \frac{\sum_{j=a_m}^{b_m} \{|S(j)| - |A_m||E(j)|\}^2}{\sum_{j=a_m}^{b_m} |S(j)|^2} \qquad (7)$$

gegeben. Wenn der NSR-Wert größer als eine vorbestimmte Schwelle von beispielsweise 0,3 ist, das heißt, wenn der Fehler größer ist, kann geschlossen werden, dass die Annäherung von $|S(j)|$ durch $|A_m|$ $|E(j)|$ für das Band nicht gut ist, das heißt, das Erregungssignal $|E(j)|$ ist nicht als die Basis geeignet, so dass das Band als UV (stimmlos) festgestellt wird. Andernfalls kann geschlossen werden, dass die Näherung akzeptabel ist, so dass das Band als V (stimmhaft) festgestellt wird.

[0075] Einem Amplituden-Wiederauswertungsabschnitt **108** werden vom Orthogonaltransformationsabschnitt **105** Daten auf der Frequenzachse, vom Feintonhöhensuchabschnitt **106** Daten der Amplitude $|A_m|$, die ausgewertet werden, um Feintonhöhendaten zu sein, und aus dem V/W-Unterscheidungsabschnitt **107** die V/W-Unterscheidungsdaten zugeführt. Der Amplituden-Wiederauswertungsabschnitt **108** findet wieder die Amplitude für das Band, das vom V/W-Unterscheidungsabschnitt **107** als stimmlos (UV) festgestellt worden ist. Die Amplitude $|A_m|_{UV}$ für dieses W-Band kann durch

$$|A_m|_{UV} = \sqrt{\sum_{j=a_m}^{b_m} |S(j)|^2 / (b_m - a_m + 1)} \qquad (8)$$

gefunden werden.

[0076] Daten aus dem Amplituden-Wiederauswertungsabschnitt **108** werden einem Datenzahl-Umwandlungsabschnitt **109** zugeführt, der ein Abschnitt zur Ausführung einer mit einer Abtastratenumwandlung vergleichbaren Verarbeitung ist. Der Datenzahl-Umwandlungsabschnitt **109** sorgt für eine konstante Zahl von Daten hinsichtlich der Änderungen der Zahl geteilter Bänder auf der Frequenzachse und folglich entsprechend der Tonhöhe der Zahl von Daten, vor allem der Zahl von Amplitudendaten. Das heißt, wenn die effektive Bandbreite eingestellt ist, dass sie bis zu 3400 kHz herauf ist, wird die effektive Bandbreite entsprechend der Ton-

höhe in 8 bis 63 Bänder geteilt, und folglich wird die Zahl m_{MX} + 1 der Daten der Amplitude $|A_m|$ (einschließlich der Amplitude $|A_m|_{UV}$ des W-Bandes) in einem Bereich von 8 bis 63 geändert. Folglich wandelt der Datenzahl-Umwandlungsabschnitt **109** die variable Zahl m_{MX} + 1 in Daten einer vorbestimmten Zahl N_C, beispielsweise 44, um.

[0077] Bei der vorliegenden Ausführungsform werden Leer- bzw. Dummydaten, die den Wert von den letzten Daten in einem Block zu den ersten Daten im Block interpolieren, zu den Amplitudendaten für den Block eines einzelnen effektiven Bandes auf der Frequenzachse addiert, um die Zahl von Daten auf NF zu erweitern. Die resultierenden Daten werden durch eine bandbreitenbegrenzungsartige Überabtastung mit einem Überabtastungsfaktor K_{OS}, beispielsweise 8, verarbeitet, um Amplitudendaten zu finden, deren Zahl K_{OS}-mal die Zahl der Amplitudendaten vor der Verarbeitung ist. Die Zahl gleich $((m_{MX} + 1) \times K_{OS})$ der Amplitudendaten wird zur Erweiterung auf eine noch größere Zahl N_M, beispielsweise 2048, direkt interpoliert, und die N_M Einheiten von Daten werden zur Umwandlung in die oben erwähnte vorbestimmte Zahl N_C von Daten, beispielsweise 44, subabgetastet.

[0078] Daten aus dem Datenzahl-Umwandlungsabschnitt **109**, das heißt die oben erwähnten M Einheiten der Amplitudendaten werden zu einem Vektorquantisierungsabschnitt **110** übertragen, bei dem die Daten in Datengruppen gruppiert werden, deren jede aus einer vorbestimmten Zahl von Daten besteht. Die Daten in jeder dieser Datengruppe werden in einem Vektor wiedergegeben und vektorquantisiert. Quantisierte Ausgangsdaten aus dem Vektorquantisierungsabschnitt **110** werden an einem Ausgangsanschluss **111** ausgegeben. Feintonhöhendaten aus dem Feintonhöhensuchabschnitt **106** werden von einem Tonhöhencodierer **115** codiert und über einen Ausgangsanschluss **112** ausgegeben.

[0079] Die Stimmhaft/Stimmlos-Unterscheidungsdaten (V/UV-Daten) aus dem Stimmhaft/Stimmlos-Ton-Unterscheidungsabschnitt 107 werden über einen Ausgangsanschluss **113** ausgegeben. Es sei darauf hingewiesen, dass die V/UV-Unterscheidungsdaten aus dem V/UV-Unterscheidungsabschnitt 107 Daten (V/UV-Code) sein können, die den Grenzpunkt zwischen dem stimmhaften Bereich und dem stimmlosen Bereich für alle Bänder darstellen, deren Zahl auf etwa 12 reduziert worden ist. Die Daten aus den Ausgangsanschlüssen **111** bis **113** werden als Signale eines vorbestimmten Übertragungsformats übertragen.

[0080] Diese Daten werden durch Verarbeitung von Daten in jedem aus der Zahl Abtastwerte N, beispielsweise 256 Abtastwerte bestehenden Block erzeugt. Da jedoch die Blöcke auf der Zeitachse mit dem aus L Abtastwerten als eine Einheit bestehenden Rahmen vorgeschoben werden, können die übertragenen Daten auf der Basis der Rahmen als Einheiten erzeugt werden. Das heißt, die Tonhöhendaten, V/UV-Entscheidungsdaten und die Amplitudendaten werden mit einem rahmenbasierten Zyklus aktualisiert.

[0081] Bezüglich **Fig.** 6 wird eine schematische Anordnung der Synthetisierungsseite (Decodierungsseite) zum Synthetisieren von Sprachsignalen auf der Basis der übertragenen Daten erläutert.

[0082] Bezüglich **Fig.** 6 werden die oben erwähnten vektorquantisierten Amplitudendaten, die codierten Tonhöhendaten und die V/W-Entscheidungsdaten an Eingangsanschlüssen **121**, **122** bzw. 123 eingegeben. Die quantisierten Amplitudendaten aus dem Eingangsanschluss **121** werden einem inversen Vektorquantisierungsabschnitt **124** zur inversen Quantisierung zugeführt und dann einem inversen Datenzahlumwandlungsabschnitt **125** zur inversen Umwandlung zugeführt. Der inverse Datenzahlumwandlungsabschnitt **125** führt eine Komplementoperation der vom Datenzahlumwandlungsabschnitt **109** ausgeführten Datenzahlumwandlung aus, und resultierende Amplitudendaten werden zu einem Stimmhafttonsyntheseabschnitt **126** und einem Stimmlostonsyntheseabschnitt **127** übertragen. Codierte Tonhöhendaten vom Eingangsanschluss **122** werden von einem Tonhöhendecodierer **128** decodiert und dann zum inversen Datenzahlumwandlungsabschnitt 125, zum Stimmhafttonsyntheseabschnitt **126** und zum Stimmlostonsyntheseabschnitt **127** übertragen. Die V/W-Entscheidungsdaten vom Eingangsanschluss **123** werden zum Stimmhafttonsyntheseabschnitt **126** und zum Stimmlostonsyntheseabschnitt **127** übertragen.

[0083] Der Stimmhafttonsyntheseabschnitt **126** synthetisiert eine Stimmhafttonwellenform auf der Zeitachse durch beispielsweise eine Cosinuswellensynthese, und der Stimmlostonsyntheseabschnitt **127** synthetisiert eine Stimmlostonwellenform durch Filterung beispielsweise des Weißrauschens mit einem Bandpassfilter. Die resultierende Stimmhafttonwellenform und resultierende Stimmlostonwellenform werden durch einen Addierer **129** summiert, um von einem Ausgangsanschluss **130** ausgegeben zu werden. In diesem Fall werden die Amplitudendaten, die Tonhöhendaten und die V/W-Entscheidungsdaten für jeden aus L Einheiten von beispielsweise 160 Abtastwerten aktualisiert. Jedoch zur Verbesserung einer Interrahmenkontinuität oder – glattheit werden die Werte der Amplitudendaten und der Tonhöhendaten zu Datenwerten in beispielsweise den Mittelabschnitten in einem einzelnen Rahmen gemacht, und Datenwerte bis zur mittleren Position des nächsten Rahmens (ein einzelner Rahmen während der Synthese) werden durch Interpolation gefunden. Das heißt, beispielsweise in einem einzelnen Rahmen während der Synthese von der Mitte des Rahmens zur Analyse bis zur Mitte des nächsten Rahmens zur Analyse werden Datenwerte am Startabtastpunkt und die am Anschluss- bzw. Endeabtastpunkt (oder am Startpunkt des nächsten Syntheserahmens) erzeugt, und Datenwerte zwischen diesen Abtastpunkten werden durch Interpolation gefunden.

[0084] Wenn andererseits der oben erwähnte V/UV-Code als V/W-Entscheidungsdaten übertragen wird, kön-

nen alle Bänder in einem Grenzpunkt entsprechend dem V/UV-Code in den stimmhaften Tonbereich (V-Bereich) und den stimmlosen Tonbereich (W-Bereich) geteilt werden, und die V/W-Entscheidungsdaten können entsprechend der Abgrenzung erzeugt werden. Es ist eine Selbstverständlichkeit, dass bei Reduzierung der Zahl Bänder auf der Syntheseseite (Codiererseite) auf eine vorbestimmte Zahl von beispielsweise 12 Bändern die Zahl der Bänder in der mit der ursprünglichen Tonhöhe übereinstimmenden variablen Zahl gelöst oder wiedergewonnen werden können.

[0085] Es wird die Syntheseverarbeitung durch den Stimmhafttonsyntheseabschnitt **126** detailliert erläutert.

[0086] Wenn der stimmhafte Ton für einen einzelnen Syntheserahmen (aus L Abtastwerten, beispielsweise 160 Abtastwerte) auf der Zeitachse des als der stimmhafte Ton unterschiedenen m-ten Bandes gleich $V_m(n)$ ist, kann er durch

$$V_m(n) = A_m(n) \cos(\theta_m(n)) \quad 0 \le n < L \tag{9}$$

ausgedrückt werden, wobei der Zeitindex (Abtastwertzahl) im Syntheserahmen verwendet wird. Die stimmhaften Töne aller als stimmhafte Töne unterschiedenen Bänder werden summiert ($\Sigma V_m(n)$), um einen entgültigen stimmhaften Ton V(n) zu synthetisieren.

[0087] In der obigen Formel (9) ist $A_m(n)$ die Amplitude der vom Startrand bis zum Anschluss- bzw. Enderand des Syntheserahmens interpolierten m-ten Oberschwingungen ist. Am einfachsten reicht es aus, den Wert der m-ten Oberschwingungen der auf der rahmenweisen Basis aktualisierten Amplitudendaten zu interpolieren. Das heißt, es reicht aus, $A_m(n)$ aus der folgenden Formel

$$A_m(n) = (L - n)A_{0M}/L + nA_{Lm}/L \tag{10}$$

zu berechnen, wobei A_{0m} der Amplitudenwert der m-ten Oberschwingungen auf dem Startrand (n = 0) des Syntheserahmens ist und A_{Lm} der Amplitudenwert der m-ten Oberschwingungen des Enderandes des Syntheserahmens (n = L: auf dem Startrand des nächsten Syntheserahmens) ist.

[0088] Die Phase $\theta_m(n)$ in der obigen Formel (9) kann durch

$$\theta_m(n) = m\omega_{01}n + n^2m(\omega_{L1} - \omega_{01})/2L + \Phi_{0m} + \Delta\omega n \tag{11}$$

gefunden werden, wobei Φ_{0m} die Phase der m-ten Oberschwingungen auf dem Startrand des Syntheserahmens (n = 0) (oder die Anfangsphase bzw. initiale Phase des Rahmens) ist, und ω_{01} die fundamentale Winkelfrequenz auf dem Startrand des Syntheserahmens (n = 0) ist. ω_{L1} ist die fundamentale Winkelfrequenz auf dem Enderand des nächsten Syntheserahmens (n = L). $\Delta\omega$ in der obigen Formel (11) ist minimal eingestellt, so dass die Phase Φ_{LM} für n = L gleich $\theta_m(L)$ ist.

[0089] Die Art und Weise, wie die Amplitude $A_m(n)$ und die Phase $\theta_m(n)$ für ein beliebiges m-tes Band entsprechend den Ergebnissen der V/UV-Unterscheidung für n = 0 und n = L gefunden wird, wird nachfolgend erläutert.

[0090] Wenn das m-te Band sowohl für n = 0 als auch n = L ein stimmhafter Ton ist, kann die Amplitude $A_m(n)$ durch lineare Interpolation der übertragenen Amplitudenwerte A_{0m} und A_{Lm} aus der obigen Formel (10) berechnet werden. Was die Phase $\theta_m(L)$ betrifft, wird $\theta\omega$ so eingestellt, dass $\theta_m(0) = \Phi_{0m}$ für n = 0 und $\theta_m(L) = \Phi_{Lm}$ für n = L gilt.

[0091] Wenn für n = 0 der Ton V (stimmhaft) ist und für n = L gleich W (stimmlos) ist, wird die Amplitude $A_m(n)$ linear interpoliert, so dass die Amplitude $A_m(0)$ bei $A_m(L)$ aus der übertragenen Amplitude A_{0m} für $A_m(0)$ gleich 0 wird. Der übertragene Amplitudenwert A_{Lm} für n = L ist der Amplitudenwert für den stimmlosen Ton und wird, wie später erläutert, zum Synthetisieren des stimmlosen Tones angewendet. Die Phase $\theta_m(n)$ wird so eingestellt, dass $\theta_m(0) = \Phi_{0m}$ und $\theta\omega = 0$ gilt .

[0092] Wenn für n = 0 der Ton UV (stimmlos) und für n = L stimmhaft (V) ist, wird die Amplitude $A_m(n)$ linear interpoliert, so dass die Amplitude $A_m(0)$ für n = 0 gleich 0 ist und gleich der übertragenen Amplitude ALm für n = L wird. Was die Phase $\theta_m(n)$ betrifft, die den Phasenwert θ_{Lm} auf dem Enderand des Rahmens als die Phase $\theta_m(0)$ für n = 0 verwendet, so ist $\theta_m(0)$ durch

$$\theta_m(0) = \theta_{Lm} - m(\omega_{01} + w_{L1})L/2 \tag{12}$$

ausgedrückt, wobei $\Delta\omega = 0$ gilt.

[0093] Es wird die Technik des Einstellens von $\Delta\theta$ so, dass $\theta_m(L)$ gleich Φ_{Lm} ist, wenn sowohl für n = 0 als auch n = L der Ton V (stimmhaft) ist, erläutert. Durch Setzen von n = L in der obigen Formel (11) wird die folgende Formel

$$\theta_m(L) = m\omega_{01}L + L^2m(\omega_{L1} - \omega_{01})/2L + \Phi_{0m} + \Delta\omega L$$
$$= m(\omega_{01} + \omega_{L1})L/2 + \Phi_{0m} + \Delta\omega L$$
$$= \Phi_{Lm}$$

erhalten. Die obige Formel kann so geordnet werden, dass sie

$$\Delta\omega = (\mathrm{mod}2\pi((\Phi_{Lm} - \Phi_{0m}) - mL(\omega_{01} + \omega_{L1})/2)/L \quad (13)$$

ergibt, wobei mod2π(x) eine Funktion ist, die den Hauptwert von x zwischen –π und +π zurückbringt. Wenn beispielsweise x = 1,3π gilt, so gilt mod2π(x) = –0,7π. Wenn x = 2,3π gilt, so gilt mod2π (x) = 0,3π, und wenn x = –1,3π gilt, gilt mod2π (x) = 0,7π.

[0094] Die **Fig.** 7A zeigt ein Beispiel eines Spektrums stimmhafter Signale, wobei die Bänder mit den Bandzahlen (Oberschwingungszahlen) von 8, 9 und 10 von W-Tönen (stimmlosen Tönen) und die verbleibenden Bänder von V-Tönen (stimmhaften Tönen) sind. Die Zeitachsensignale der Bänder der V-Töne werden vom Stimmhafttonsyntheseabschnitt **126** synthetisiert, und die Zeitachsensignale der Bänder der UV-Töne werden vom Stimmlostonsyntheseabschnitt **127** synthetisiert.

[0095] Wenn jedoch der stimmhafte Bandbereich (V-Bandbereich) und der stimmlose Bandbereich (UV-Bandbereich) anders voneinander abgegrenzt sind als von einem einzigen Punkt, kann der übertragene V/UV-Code auf 7 gesetzt werden, während alle anderen Bänder mit m nicht kleiner als 8 als stimmloser Bandbereich gemacht werden können. Alternativ dazu kann der V/W-Code, der alle Bänder V (stimmhaft) macht, übertragen werden.

[0096] Es wird die Operation der Synthetisierung von W-Tönen durch den W-Tonsyntheseabschnitt **127** erläutert.

[0097] Die Weißrauschensignalform auf der Zeitachse von einem Weißrauschengenerator **131** wird mit einer geeigneten Fensterfunktion (beispielsweise ein Hammingfenster) bei einer vorbestimmten Länge (beispielsweise 256 Abtastwerte) multipliziert und von einem STFT-Prozessor **132** mit einer Kurztermfouriertransformation (= STFT) verarbeitet, wodurch ein Leistungsspektrum des Weißrauschens auf der Frequenzachse wie bei B in **Fig.** 7 gezeigt erzeugt wird. Das Leistungsspektrum aus dem STFT-Prozessor **132** wird zu einem Bandpassfilter **133** übertragen, wo das Spektrum mit der Amplitude $|A_m|_{UV}$ für die UV-Bänder (beispielsweise m = 8, 9 oder 10) multipliziert wird, wie es bei C in **Fig.** 7 gezeigt ist, während die Amplitude der V-Bänder auf 0 gesetzt wird. Dem Bandpassfilter **133** werden auch die oben erwähnten Amplitudendaten, Tonhöhendaten und V/UV-Entscheidungsdaten zugeführt.

[0098] Da der V/UV-Code, der nur einen einzelnen Grenzpunkt zwischen dem stimmhaften Bereich (V-Bereich) und dem stimmlosen Bereich (W-Bereich) aller Bänder bezeichnet, als die V/UV-Entscheidungsdaten verwendet wird, werden die Bänder in Richtung zur niedrigeren Frequenz des bezeichneten Grenzpunktes als die stimmhaften Bänder (V-Bänder) gesetzt, und die Bänder in Richtung zur höheren Frequenz des bezeichneten Grenzpunktes werden als die stimmlosen Bänder (UV-Bänder) gesetzt. Die Zahl dieser Bänder kann auf eine vorbestimmte kleinere Zahl, beispielsweise 12, reduziert werden.

[0099] Ein Ausgangssignal aus dem Bandpassfilter **133** wird einem ISTFT-Prozessor **134** zugeführt, während die Phase mit einer inversen STFT-Verarbeitung unter Verwendung der Phase des ursprünglichen Weißrauschens zur Umwandlung in Signale auf der Zeitachse verarbeitet wird. Ein Ausgangssignal aus dem ISTFT-Prozessor **134** wird zu einem Überlapp- und Addierabschnitt **135** übertragen, wo eine Überlappung und Addition mit einer geeigneten Gewichtung auf der Zeitachse wiederholt ausgeführt wird, um die Wiederherstellung der ursprünglichen kontinuierlichen Rauschwellenform zu ermöglichen, wodurch die kontinuierliche Wellenform auf der Zeitachse synthetisiert wird. Ein Ausgangssignal aus dem Überlapp- und Addierabschnitt **135** wird dem Addierer **129** zugeführt.

[0100] Die auf diese Weise in den Syntheseabschnitten **126**, **127** synthetisierten und als die Zeitachsensignale wiederhergestellten V- und UV-Signale werden vom Addierer **129** mit einer festen Mischrate summiert, und dann werden die wiedergegebenen Signale vom Ausgangsanschluss **130** ausgegeben.

[0101] Indessen können die Anordnung der in **Fig.** 1 gezeigten Sprachanalyseseite (Codiererseite) und die Anordnung der in **Fig.** 6 gezeigten Sprachsyntheseseite (Decodiererseite), die als Hardwarekomponenten beschrieben worden sind, auch durch ein Softwareprogramm realisiert werden, das einen Digitalsignalprozessor (DSP) verwendet.

[0102] Als nächstes werden unter Bezugnahme auf die Zeichnungen konkrete Beispiele jedes Teils und Abschnitts des oben erwähnten Synthese-Analyse-Codierers oder Vocoders für Sprachsignale detailliert erläutert.

[0103] Zuerst wird ein konkretes Beispiel eines Tonhöhenextraktionsverfahrens durch den in **Fig.** 1 gezeigten Tonhöhenextraktionsabschnitt **103**, das heißt ein konkretes Beispiel eines Tonhöhenextraktionsverfahrens zur Extraktion der Tonhöhe aus der stimmhaften Eingangssignalwellenform erläutert.

[0104] Die Sprachtöne werden in stimmhafte Töne und stimmlose Töne geteilt. Die stimmlosen Töne, die Töne ohne Schwingungen der Stimmbänder sind, werden als nicht periodisches Rauschen beobachtet. Normalerweise sind die Majorität von Sprachtönen stimmhafte Töne, und die stimmlosen Töne sind besondere Konsonanten, die als stimmlose Konsonanten bezeichnet werden. Die Periode der stimmhaften Töne wird durch die Periode von Schwingungen der Stimmbänder bestimmt und als eine Tonhöhenperiode bezeichnet, deren Kehrwert als Tonhöhenfrequenz bezeichnet wird. Die Tonhöhenperiode und die Tonhöhenfrequenz sind wichtige Determinanten der Höhe und Intonation von Stimmen bzw. Sprachen. Deshalb ist unter den Prozessen der Sprachsynthese zum Analysieren und Synthetisieren von Sprachen eine exakte, nachfolgend als Tonhöhenextraktion bezeichnete Extraktion der Tonhöhenperiode der ursprünglichen Sprachwellenform wichtig.
[0105] Das oben erwähnte Tonhöhenextraktionverfahren wird als Wellenformverarbeitungsverfahren zum Detektieren wird eingeteilt in die Kategorien Wellenformverarbeitungsverfahren zum Detektieren der Spitze der Periode auf der Wellenform, Korrelationsverarbeitungsverfahren, welches die Stärke der Korrelationsverarbeitung auf die Wellenformverzerrung verwendet, und Spektrumverarbeitungsverfahren, welches eine periodische Frequenzstruktur des Spektrums verwendet.
[0106] Ein Autokorrelationsverfahren, das eines der Korrelationsverfahren ist, wird unter Bezugnahme auf die **Fig.** 8 erläutert. Die **Fig.** 8A zeigt eine Eingangssprachtonwellenform x(n) für 300 Abtastwerte, und die **Fig.** 8B zeigt eine Wellenform, die durch Finden einer Autokorrelationsfunktion des in **Fig.** 8A gezeigten x(n) erzeugt wird. Die **Fig.** 8C zeigt eine Wellenform C[x(n)], die durch ein Mitteabschneiden bzw. Mitteclipping bei einem in **Fig.** 8A gezeigten Abschneide- bzw. Clippingpegel CL erzeugt wird, und **Fig.** 8D zeigt eine Wellenform Rc(k) die durch Finden der Autokorrelation des in **Fig.** 8C gezeigten C[x(n)] erzeugt wird.
[0107] Die Autokorrelationsfunktion der in **Fig.** 8A gezeigten Eingangssprachwellenform x(n) für 300 Abtastwerte ergibt sich wie oben beschrieben als eine in **Fig.** 8B gezeigte Wellenform Rx(k). Bei der Wellenform Rx(k) der in **Fig.** 8B gezeigten Autokorrelationsfunktion wird bei der Tonhöhenperiode eine starke Spitze gefunden. Jedoch wird auch eine Zahl exzessiver Spitzen aufgrund von Dämpfungsschwingungen der Stimmbänder beobachtet. Zur Reduzierung dieser exzessiven Spitzen ist es denkbar, die Autokorrelationsfunktion von der in **Fig.** 8C gezeigten Mitteabschneide- bzw. Mitteclipwellenform C[x(n)] zu finden, bei der die Wellenform, die im absoluten Wert kleiner als der in **Fig.** 8A gezeigte Clippingpegel ± CL ist, unterdrückt ist. In diesem Fall bleiben in der in **Fig.** 8C gezeigten, in der Mitte abgeschnittenen bzw. geclipten Wellenform C[x(n)] nur mehrere Impulse beim ursprünglichen Tonhöhenintervall, und in der daraus gefundenen Wellenform der Autokorrelationsfunktion Re (k) sind exzessive Spitzen reduziert.
[0108] Die durch die obige Tonhöhenextraktion erhaltene Tonhöhe ist, wie oben beschrieben, eine wichtige Determinante der Höhe und Internotation von Stimmen. Die präzise Tonhöhenextraktion aus der ursprünglichen Stimmenwellenform ist beispielsweise für eine hocheffiziente Codierung von Stimmenwellenformen angenommen.
[0109] Indessen ist beim Finden der Tonhöhe aus der Spitze der Autokorrelation der Eingangssprachsignalwellenform der Clippingpegel konventionell so eingestellt worden, dass die Spitze durch das mittige Clipping als scharf erscheinend gefunden wird. Speziell ist der Clippingpegel so niedrig eingestellt worden, dass das Fehlen des Signals eines winzigen Pegels aufgrund des Clippings vermieden ist.
[0110] Wenn demgemäss scharfe Fluktuationen des Eingangspegels, beispielsweise ein Einstellen des Sprachtons mit dem niedrigen Clippingpegel vorhanden ist, werden zu dem Zeitpunkt exzessive Spitzen erzeugt, bei dem der Eingangspegel erhöht ist. Infolgedessen wird der Effekt des Clippings kaum erhalten, wobei die Gefahr einer Instabilität der Tonhöhenextraktion zurückbleibt.
[0111] Infolgedessen wird nachfolgend ein erstes konkretes Beispiel des Tonhöhenextraktionsverfahrens erläutert, bei dem eine sichere Tonhöhenextraktion auch dann möglich ist, wenn der Pegel der Eingangssprachwellenform in einem einzelnen Rahmen scharf geändert wird.
[0112] Das heißt, bei dem ersten Beispiel des Tonhöhenextraktionsverfahrens wird die einzugebende Sprachsignalwellenform auf der blockweisen Basis ausgegeben. Bei dem Tonhöhenextraktionsverfahren zur Extraktion der Tonhöhe auf der Basis des zentral bzw. mittig geklippten Ausgangssignals wird der Block in mehrere Subblöcke geteilt, um einen Pegel zum Clipping jedes der Subblöcke zu finden, und beim mittigen Abschneiden bzw. Clippen des Eingangssignals wird der Clippingpegel im Block auf der Basis des für jeden der Subblöcke gefundenen Pegels zum Clipping geändert.
[0113] Auch wenn es eine große Fluktuation des Spitzenpegels zwischen benachbarten Subblöcken unter den mehreren Subblöcken in dem Block gibt, wird der Clippingpegel beim mittigen Clipping im Block geändert.
[0114] Der Clippingpegel beim mittigen Clippling kann im Block stufenweise oder kontinuierlich geändert werden.
[0115] Gemäß diesem ersten Beispiel des Tonhöhenextraktionsverfahrens wird die auf der blockweisen Basis ausgegebene Eingangssprachsignalwellenform in mehrere Subblöcke geteilt, und der Clippingpegel wird innerhalb des Blocks auf der Basis des für jeden der Subblöcke gefundenen Pegels zum Clipping geändert, wodurch eine sichere Tonhöhenextraktion ausgeführt wird.
[0116] Außerdem wird beim Vorhandensein einer großen Fluktuation des Spitzenpegels zwischen benach-

barten Subblöcken unter den mehreren Subblöcken der Clippingpegel innerhalb des Blocks geändert, wodurch eine sichere Tonhöhenextraktion realisiert wird.

[0117] Das erste konkrete Beispiel des Tonhöhenextraktionsverfahrens wird unter Bezugnahme auf die Zeichnungen erläutert.

[0118] Die **Fig.** 9 ist ein funktionelles Blockschaltbild zur Darstellung der Funktion der vorliegenden Ausführungsform des Tonhöhenextraktionsverfahrens gemäß der vorliegenden Erfindung.

[0119] Bezüglich **Fig.** 9 sind bei diesem Beispiel vorgesehen: ein Blockextraktionsverarbeitungsabschnitt **10** zum Ausgeben eines von einem Eingangsanschluss **1** zugeführten Eingangssprachsignals auf der blockweisen Basis, ein Clippingpegeleinstellungsabschnitt **11** zum Einstellen des Clippingpegels von einem einzelnen Block des vom Blockextraktionsverarbeitungsabschnitts **10** extrahierten Eingangssprachsignals, ein Mitteclipverarbeitungsabschnitt **12** zum mittigen Clipping eines einzelnen Blocks des Eingangssprachsignals bei dem vom Clippingpegeleinstellabschnitt **11** eingestellten Clippingpegel, ein Autokorrelationsberechnungsabschnitt **13** zur Berechnung einer Autokorrelation von der Mitteclipwellenform aus dem Mitteclipverarbeitungsabschnitt **12**, und ein Tonhöhenkalkulator **14** zur Berechnung der Tonhöhe aus der Autokorrelationswellenform aus dem Autokorrelationsberechnungsabschnitt **13**.

[0120] Der Clippingpegeleinstellungsabschnitt **11** weist auf: einen Subblockteilungsabschnitt **15** zur Teilung eines einzelnen Blocks des vom Blockextraktionsabschnitt **10** zugeführten Eingangssprachsignals in mehrere Subblöcke (bei der vorliegenden Ausführungsform zwei Subblöcke, das heißt eine erste und letzte Hälfte), eine Spitzenpegelextraktionseinheit **16** zur Extraktion des Spitzenpegels sowohl im ersten halben als auch letzten halben Subblock des zum Subblockteilungsabschnitt **15** geteilten Eingangssprachsignals, einen Maximumspitzenpegeldetektionsabschnitt **17** zum Detektieren des Maximumspitzenpegels in der ersten und letzten Hälfte von dem vom Spitzenpegelextraktionsabschnitt **16** extrahierten Spitzenpegel, einen Komparator **18** zum Vergleichen des Maximumspitzenpegels in der ersten Hälfte und des Maximumspitzenpegels in der letzten Hälfte aus dem Maximumspitzenpegeldetektorabschnitt **17** unter gewissen Bedingungen, und einen Clippingspegelsteuerungsabschnitt **19** zum Einstellen des Clippingspegels aus Ergebnissen des Vergleichs durch den Komparator **18** und der zwei vom Maximumspitzenpegeldetektorabschnitt **17** detektierten Maximumspitzenpegel und zur Steuerung des zentralen Clipverarbeitungsabschnitt **12**.

[0121] Der Spitzenpegelextraktionsabschnitt **16** ist durch Subblockspitzenpegelextraktionsabschnitte **16a**, **16b** gebildet. Der Subblockspitzenpegelextraktionsabschnitt **16a** extrahiert den Spitzenpegel aus der durch Teilung des Blocks durch den Subblockteilungsabschnitt **15** erzeugten ersten Hälfte. Der Subblockspitzenpegelextraktionsabschnitt **16b** extrahiert den Spitzenpegel aus der durch Teilung des Blocks durch den Subblockteilungsabschnitt **15** erzeugten letzten Hälfte.

[0122] Der Maximumspitzenpegeldetektionsabschnitt **17** ist durch Subblockmaximumspitzenpegeldetektoren **17a**, **17b** gebildet. Der Subblockmaximumspitzenpegeldetektor **17a** detektiert den Maximumspitzenpegel der ersten Hälfte aus dem vom Subblockspitzenpegelextraktionsabschnitt **16a** extrahierten Spitzenpegel der ersten Hälfte. Der Subblockmaximimspitzenpegeldetektor **17b** detektiert den Maximumspitzenpegel der letzten Hälfte aus dem vom Subblockspitzenpegelextraktionsabschnitt **16b** extrahierten Spitzenpegel der letzten Hälfte.

[0123] Als nächstes werden eine Operation der aus dem in **Fig.** 9 gezeigten funktionellen Block gebildeten vorliegenden Ausführungsform unter Bezugnahme auf ein in **Fig.** 10 gezeigtes Flussdiagramm und eine in **Fig.** 11 gezeigte Wellenform erläutert.

[0124] Zuerst wird im Flussdiagramm der **Fig.** 10 beim in Gang bringen der Operation beim Schritt S1 eine Eingangssprachsignalwellenform auf der blockweisen Basis ausgegeben. Insbesondere wird das Eingangssprachsignal mit einer Fensterfunktion multipliziert und an dem Eingangssprachsignal eine partielle Überlappung ausgeführt, um die Eingangssprachsignalwellenform auszuschneiden. Infolgedessen wird die in **Fig.** 11A gezeigte Eingangssprachsignalwellenform eines einzelnen Rahmens (**256** Abtastwerte) erzeugt. Dann geht die Operation zum Schritt S2 vor.

[0125] Beim Schritt S2 wird ein einzelner Block des beim Schritt S1 ausgegebenen Eingangssprachsignals weiter in mehrere Subblöcke geteilt. Beispielsweise wird bei der in **Fig.** 11A gezeigten Eingangssprachsignalwellenform eines einzelnen Blocks die erste Hälfte auf n = 0, 1, ..., 127 gesetzt, und die letzte Hälfte wird auf n = 128, 129, ..., 255 gesetzt. Dann geht die Operation zum Schritt S3 vor.

[0126] Beim Schritt S3 werden Spitzenpegel der Eingangssprachsignale in der beim Schritt S2 durch Teilung erzeugten ersten und letzten Hälfte extrahiert. Diese Extraktion ist die Operation des in **Fig.** 9 gezeigten Spitzenpegelextraktionsabschnitts **16**.

[0127] Beim Schritt S4 werden Maximumspitzenpegel P_1 und P_2 in den jeweiligen Subblöcken aus den beim Schritt S3 extrahierten Spitzenpegeln in der ersten und letzten Hälfte detektiert. Diese Detektion ist die Operation des in **Fig.** 9 gezeigten Maximumspitzenpegeldetektionsabschnitts **17**.

[0128] Beim Schritt S5 werden die beim Schritt S4 jeweils detektierten Maximumspitzenpegel P_1 und P_2 in der ersten und letzten Hälfte unter gewissen Bedingungen miteinander verglichen, und es wird eine Detektion ausgeführt, ob die Pegelfluktuation der Eingangssprachsignalwellenform in einem einzelnen Rahmen scharf ist

oder nicht. Die erwähnten Bedingungen sind hier, dass der Maximumspitzenpegel P_1 der ersten Hälfte kleiner ist als der vom Maximumspitzenpegel P_2 der letzten Hälfte durch Multiplikation mit einem Koeffizienten k ($0 < k < 1$) erzeugte Wert ist, oder dass der Maximumspitzenpegel P_2 der letzten Hälfte kleiner als der vom Maximumspitzenpegel P_1 der ersten Hälfte durch Multiplikation mit einem Koeffizienten k ($0 < k < 1$) erzeugte Wert ist. Demgemäss werden bei diesem Schritt S5 die Maximumspitzenpegel P_1 und P_2 der ersten bzw. letzten Hälfte miteinander auf die Bedingung $P_1 < k \cdot P_2$ oder $k \cdot P_1 > P_2$ hin verglichen. Dieser Vergleich ist die Operation des in **Fig.** 9 gezeigten Komparators **18**. Als Ergebnis des Vergleichs der Maximumspitzenpegel P_1 und P_2 der ersten bzw. letzten Hälfte unter den oben erwähnten Bedingun- gen beim Schritt S5 geht die Operation bei der Feststellung, dass die Pegelfluktuation des Eingangssprachsignals groß ist (JA) zum Schritt S6 vor. Wenn festgestellt wird, dass die Pe- gelfluktuation des Eingangssprachsignals nicht groß ist (NEIN), geht die Operation zum Schritt S7 vor.

[0129] Beim Schritt 56 wird entsprechend dem Ergebnis der Entscheidung beim Schritt S5, dass die Fluktuation des Maximumpegels groß ist, eine Berechnung mit verschiedenen Clippingpegeln ausgeführt. In der **Fig.** 11B zum Beispiel sind der Clippingpegel in der ersten Hälfte ($0 < n < 127$) und der Clippingpegel in der letzten Hälfte ($128 < n < 255$) auf $k \cdot P_1$ bzw. $k \cdot P_2$ eingestellt.

[0130] Andererseits wird beim Schritt S7 entsprechend dem Ergebnis der Entscheidung beim Schritt S5, dass die Pegelfluktuation des Eingangssprachsignals in einem Block nicht groß ist, eine Berechnung mit einem einheitlichen Clippingpegel ausgeführt. Beispielsweise wird vom Maximumspitzenpegel P_1 und Maximumspitzenpegel P_2 der kleinere mit k multipliziert, um $k \cdot P_1$ oder $k \cdot P_2$ zu erzeugen. $k \cdot P_1$ oder $k \cdot P_2$ wird dann abgeschnitten bzw. geclipt und gesetzt.

[0131] Diese Schritte S6 und S7 sind Operationen der in **Fig.** 9 gezeigten Clippingpegelsteuerungseinheit **19**.

[0132] Beim Schritt S8 wird eine Mitteclipverarbeitung eines einzelnen Blocks der Eingangssprachwellenform bei einem beim Schritt S6 oder S7 eingestellten Clippingpegel ausgeführt. Die Mitteclipverarbeitung ist die Operation des in **Fig.** 9 gezeigten Mitteclipverarbeitungsabschnitts **12**. Dann geht die Operation zum Schritt S9 vor.

[0133] Beim Schritt S9 wird die Autokorrelationsfunktion aus der von der Mitteclipverarbeitung beim Schritt S8 erhaltenen Mitteclipwellenform berechnet. Diese Berechnung ist die Operation der in **Fig.** 9 gezeigten Autokorrelationsberechnungseinheit **13**. Dann geht die Operation zum Schritt S10 vor.

[0134] Beim Schritt S10 wird die Tonhöhe von der beim Schritt S9 gefundenen Autokorrelationsfunktion extrahiert. Diese Tonhöhenextraktion ist die Operation des in **Fig.** 9 gezeigten Tonhöhenberechnungsabschnitts **14**.

[0135] Die **Fig.** 11A zeigt die Eingangssprachsignalwellenform, wobei ein einzelner Block aus 256 Abtastwerten von N = 0, 1, ..., 255 besteht. In der **Fig.** 11A ist die erste Hälfte auf N = 0, 1, ..., 127 eingestellt, und die letzte Hälfte ist auf N = 128, 129, ..., 255 eingestellt. Die Maximumspitzenpegel des Absolutwerts der Wellenform werden innerhalb von 100 Abtastwerten von N = 0, 1, ..., 99 in der ersten Hälfte bzw. innerhalb von 100 Abtastwerten von N = 156, 157, ..., 255 gefunden. Die auf diese Weise gefundenen Maximumspitzenpegel sind P_1 bzw. P_2. Wenn der Wert k wie in **Fig.** 11A gezeigt auf 0,6 für $P_1 = 1$ und $P_2 = 3$ eingestellt sind, gilt die folgende Formel

$P_1 (= 1) < k \cdot P_2 (= 1{,}8)$.

[0136] In diesem Fall wird für die große Pegelfluktuation der Eingangssprachsignalwellenform der Clippingpegel der ersten Hälfte auf $k \cdot P_1 = 0{,}6$ eingestellt, und der Clippingpegel der letzten Hälfte wird auf $k \cdot P_2 = 1{,}8$ eingestellt. Diese Clippingpegel sind in der **Fig.** 11B gezeigt. Eine mit dem Mitteclipping bei den in **Fig.** 11B gezeigten Clippingpegeln verarbeitete Wellenform ist in der **Fig.** 11C gezeigt. Die Autokorrelationsfunktion der in **Fig.** 11C gezeigten mittegeklippten Wellenform wird so genommen, dass sie wie in **Fig.** 11D gezeigt ist. Aus der **Fig.** 11D kann die Tonhöhe berechnet werden.

[0137] Der Clippingpegel beim Mitteclipverarbeitungsabschnitt **12** kann nicht nur wie oben beschrieben fortschreitend im Block geändert werden, sondern auch wie durch eine gestrichelte Linie in **Fig.** 11B gezeigt kontinuierlich.

[0138] Bei Anwendung des ersten Beispiels des Tonhöhenextraktionsverfahrens auf den in Bezug auf die **Fig.** 1 bis 7 erläuterten MBE-Vocoder wird die Tonhöhenextraktion des Tonhöhenextraktionsabschnitts **103** durch Detektieren des Spitzenpegels des Signals jedes durch Teilen des Blocks erzeugten Subblocks und fortschreitende oder kontinuierliche Änderung des Clippingpegels, wenn die Differenz der Spitzenpegel dieser Subblocks ungleich 0 ist, ausgeführt. Infolgedessen kann auch beim Vorhandensein einer scharfen Fluktuation des Spitzenpegels die Tonhöhe sicher extrahiert werden.

[0139] Das heißt, gemäss dem ersten Beispiel des Tonhöhenextraktionsverfahrens wird durch Ausgeben des Eingangssprachsignals auf der blockweisen Basis, Teilen des Blocks in mehrere Subblöcke und Ändern des Clippingpegels des mittegeclippten Signals auf der blockweisen Basis entsprechend dem Spitzenpegel für jeden der Subblöcke eine sichere Tonhöhenextraktion möglich gemacht.

[0140] Außerdem wird entsprechend dem Tonhöhenextraktionsverfahren, wenn die Fluktuation der Spitzenpegel benachbarter Subblöcke unter den mehreren Subblöcken groß ist, der Clippingpegel für jeden Block geändert. Infolgedessen wird auch beim Vorhandensein scharfer Fluktuationen, beispielsweise Anstieg und Abfall von Stimme bzw. Sprache, eine sichere Tonhöhenextraktion möglich.

[0141] Indessen ist das erste Beispiel des Tonhöhenextraktionsverfahrens nicht auf das durch die Zeichnungen gezeigte Beispiel beschränkt. Das hocheffiziente Codierungsverfahren, auf welches das erste Beispiel angewendet ist, ist nicht auf dem MBE-Vocoder beschränkt.

[0142] Andere Beispiele, d.h. das zweite und dritte Beispiel des Tonhöhenextraktionsverfahrens werden unter Bezugnahme auf die Zeichnungen erläutert.

[0143] Generell besteht bei Beobachtung der Autokorrelation des Eingangssprachsignals eine hohe Wahrscheinlichkeit, dass das Maximum der Spitzen die Tonhöhe ist. Wenn jedoch die Spitzen der Autokorrelation wegen der Pegelfluktuation des Eingangssprachsignals oder des Hintergrundrauschens nicht klar erscheinen, kann eine korrekte Tonhöhe nicht mit einer eingefangenen Tonhöhe, die ein Ganzzahligfaches größer ist, erhalten werden, oder es wird festgestellt, dass keine Tonhöhe vorhanden ist. Es ist auch denkbar, zur Vermeidung des obigen Problems einen erlaubten Bereich der Tonhöhenfluktuationen zu begrenzen. Es ist jedoch unmöglich, einer scharfen Änderung der Tonhöhe eines einzelnen Sprechers oder einem Alternieren zweier oder mehrerer Sprecher, die beispielsweise kontinuierliche Änderungen zwischen männlichen Stimmen und weiblichen Stimmen verursachen, zu folgen.

[0144] Infolgedessen wird ein konkretes Beispiel des Tonhöhenextraktionsverfahrens vorgeschlagen, bei dem die Wahrscheinlichkeit des Einfangens einer falschen Tonhöhe niedrig wird und bei dem die Tonhöhe stabil extrahiert werden kann.

[0145] Das heißt, das zweite Beispiel des Tonhöhenextraktionsverfahrens weist die Schritte auf: Abgrenzen eines Eingangssprachsignals auf der rahmenweisen Basis, Detektieren mehrerer Spitzen von Autokorrelationsdaten eines laufenden Rahmens, Finden einer Spitze unter den detektierten mehreren Spitzen des gegenwärtigen bzw. laufenden Rahmens und innerhalb eines eine vorbestimmte Relation mit einer in einem vom laufenden Rahmen verschiedenen Rahmen gefundenen Tonhöhe erfüllenden Tonhöhenbereichs und Feststellen der Tonhöhe des laufenden Rahmens auf der Basis der Position der auf die obige Art und Weise gefundenen Spitze.

[0146] Mit der hohen Zuverlässigkeit der Tonhöhe des laufenden Rahmens werden mehrere Tonhöhen des laufenden Rahmens durch die Position der Maximumspitze bestimmt, wenn das Maximum unter den mehreren Spitzen des laufenden Rahmens gleich oder größer als eine vorbestimmte Schwelle ist, und die Tonhöhe des laufenden Rahmens wird durch die Position der Spitze in dem Tonhöhenbereich bestimmt, der eine vorbestimmte Relation mit der in einem vom laufenden Rahmen verschiedenen Rahmen gefundenen Tonhöhe erfüllt, wenn die Maximumspitze kleiner als die vorbestimmte Schwelle ist.

[0147] Indessen weist das dritte Beispiel des Tonhöhenextraktionsverfahrens die Schritte auf: Abgrenzen eines Eingangssprachsignals auf der rahmenweisen Basis, Detektieren aller Spitzen aus Autokorrelationsdaten eines laufenden Rahmens, Finden einer Spitze unter allen detektierten Spitzen des laufenden Rahmens und innerhalb eines Tonhöhenbereichs, der eine vorbestimmte Relation mit einer in einem vom laufenden Rahmen verschiedenen Rahmen gefundenen Spitze erfüllt, und Feststellen der Tonhöhe des laufenden Rahmens auf der Basis der Position der auf die obige Art und Weise gefundenen Spitze.

[0148] Bei dem Prozess der Ausgabe des Eingangssprachsignals auf der rahmenweisen Basis mit entlang der Zeitachse fortschreitenden Blöcken als Einheiten wird das Eingangssprachsignal in Blöcke geteilt, deren jeder aus einer vorbestimmten Zahl N Abtastwerten, beispielsweise 256 Abtastwerte, besteht und entlang der Zeitachse in einem Rahmenintervall von L Abtastwerten, beispielsweise 160 Abtastwerte, das einen Überlappungsbereich von (N – L) Abtastwerten, beispielsweise 96 Abtastwerte, aufweist, bewegt.

[0149] Der Tonhöhenbereich, der die vorbestimmte Relation erfüllt, ist beispielsweise ein a- bis b-mal, beispielsweise 0,8- bis 1,2-mal größerer Bereich als eine feste Tonhöhe eines vorhergehenden Rahmens.

[0150] Bei Abwesenheit der fixierten Tonhöhe in dem vorhergehenden Rahmen wird eine typische Tonhöhe verwendet, die für jeden Rahmen gehalten wird und für eine Person, die das Objekt der Analyse sein soll, typisch ist, und der Ort der Tonhöhe wird unter Verwendung der Tonhöhe in dem a- bis b-fachen, beispielsweise 0,8- bis 1,2-fachen Bereich der typischen Tonhöhe verfolgt.

[0151] Außerdem wird in dem Fall, dass die Person plötzlich eine Stimme einer von der letzten Tonhöhe verschiedenen Tonhöhe erhebt, wird der Ort der Tonhöhe unter Verwendung einer Tonhöhe verfolgt, die ungeachtet der vergangenen Tonhöhe im laufenden Rahmen Tonhöhen springen oder überspringen kann.

[0152] Gemäß dem zweiten Beispiel des Tonhöhenextraktionsverfahrens kann die Tonhöhe des laufenden Rahmens auf der Basis der Position derjenigen Spitze unter den mehreren Spitzen bestimmt werden, die von den Autokorrelationsdaten des laufenden Rahmens des Eingangssprachsignals detektiert wird, das auf der rahmenweisen Basis abgegrenzt ist und sich in dem Tonhöhenbereich befindet, der die vorbestimmte Relation mit der in einem vom laufenden Rahmen verschiedenen Rahmen gefundenen Tonhöhe erfüllt. Deshalb wird die Wahrscheinlichkeit des Einfangens einer falschen Tonhöhe niedrig, und es kann eine stabile Tonhöhenex-

traktion ausgeführt werden.

[0153] Auch kann die Tonhöhe des laufenden Rahmens auf der Basis der Position derjenigen Spitze unter allen Spitzen bestimmt werden, die von den Autokorrelationsdaten des laufenden Rahmens des Eingangssprachsignals detektiert wird, das auf der rahmenweisen Basis abgegrenzt ist und in dem Tonhöhenbereich ist, der die vorbestimmte Relation mit der in einem vom laufenden Rahmen verschiedenen Rahmen gefundenen Tonhöhe erfüllt. Deshalb wird die Wahrscheinlichkeit des Einfangens einer falschen Tonhöhe niedrig, und es kann eine stabile Tonhöhenextraktion ausgeführt werden.

[0154] Außerdem wird gemäß dem dritten Beispiel des Tonhöhenextraktionsverfahrens die Tonhöhe des laufenden Rahmens durch die Position der Maximumspitze bestimmt, wenn das Maximum unter den mehreren Spitzen des laufenden Rahmens gleich oder höher als eine vorbestimmte Schwelle ist. Die Tonhöhe des laufenden Rahmens wird durch die Position der Spitze in dem Tonhöhenbereich bestimmt, der eine vorbestimmte Relation mit der in einem vom laufenden Rahmen verschiedenen Rahmen gefundenen Tonhöhe erfüllt, wenn die Maximumspitze kleiner als die vorbestimmte Schwelle ist. Deshalb wird die Wahrscheinlichkeit des Einfangens einer falschen Tonhöhe niedrig, und es kann eine stabile Tonhöhenextraktion ausgeführt werden.

[0155] Nachfolgend werden unter Bezugnahme auf die Zeichnungen konkrete Beispiele erläutert, bei denen das zweite und dritte Beispiel des Tonhöhenextraktionsverfahrens auf eine Tonhöhenextraktionseinrichtung angewendet wird.

[0156] Die **Fig.** 12 ist ein Blockschaltbild, das eine schematische Anordnung einer Tonhöhenextraktionseinrichtung zeigt, auf die das zweite Beispiel des Tonhöhenextraktionsverfahrens angewendet ist.

[0157] Die in **Fig.** 12 gezeigte Tonhöhenextraktionseinrichtung weist auf: einen Blockextraktionsabschnitt 209 zum Ausgeben einer Eingangssprachsignalwellenform auf der blockweisen Basis, einen Rahmenabgrenzungsabschnitt **210** zur Abgrenzung auf der blockweisen Basis der vom Blockextraktionsabschnitt **209** auf der blockweisen Basis ausgegebenen Eingangssprachsignalwellenform, eine Mitteclipverarbeitungseinheit 211 zum Mitteclipping der Sprachsignalwellenform eines laufenden Rahmens aus dem Rahmenabgrenzungsabschnitt **210**, einen Autokorrelationsberechnungsabschnitt **212** zur Berechnung von Autokorrelationsdaten aus der vom Mitteclipverarbeitungsabschnitt **211** mittegeclipten Sprachsignalwellenform, einen Spitzedetektionsabschnitt 213 zum Detektieren mehrerer oder aller Spitzen von den vom Autokorrelationsberechnungsabschnitt **212** berechneten Autokorrelationsdaten, einen Anderrahmentonhöhenberechnungsabschnitt **214** zur Berechnung einer Tonhöhe eines Rahmens (nachfolgend als anderer Rahmen bezeichnet), der verschieden von dem laufenden Rahmen aus dem Rahmenabgrenzungsabschnitt **210** ist, einen Vergleichs/Detektions-Abschnitt **215** zum Vergleichen der Spitzen danach, ob die vom Spitzendetektionsabschnitt **213** detektierten mehreren Spitzen in einem Tonhöhenbereich sind, der eine vorbestimmte Funktion mit der Tonhöhe des Andertonhöhenberechnungsabschnitts **212** erfüllt, und zum Detektieren von Spitzen in dem Bereich, und einen Tonhöhenentscheidungsabschnitt 216 zum Feststellen einer Tonhöhe des laufenden Rahmens auf der Basis der Position der vom Vergleichs/Detektions-Abschnitt 215 gefundenen Spitze.

[0158] Der Blockextraktionsabschnitt **209** multipliziert die Eingangssprachsignalwellenform mit einer Fensterfunktion, wobei eine partielle Überlappung der Eingangssprachsignalwellenform erzeugt wird, und schneidet die Eingangssprachsignalwellenform als einen Block von N Abtastwerten aus. Die Rahmenabgrenzungseinheit **210** grenzt auf der rahmenweisen L-Abtast wert-Basis die vom Blockextraktionsabschnitt **209** ausgegebene Signalwellenform auf der blockweisen Basis. In anderen Worten ausgedrückt gibt der Blockextraktionsabschnitt **209** das Eingangssprachsignal als eine Einheit von N Abtastwerten, die entlang der Zeitachse auf der rahmenweisen L-Abtastwert-Basis fortschreitet, aus.

[0159] Der Mitteclipverarbeitungsabschnitt **211** steuert Charakteristiken derart, dass die Periodizität der Eingangssprachsignalwellenform für einen einzelnen Rahmen aus dem Rahmenabgrenzungsabschnitt **210** fehlgeordnet bzw. gestört wird. Das heißt, es wird ein vorbestimmter Clippingpegel zur Reduzierung exzessiver Spitzen mittels einer Dämpfung von Stimmbändern vor Berechnung der Autokorrelation der Eingangssprachsignalwellenform eingestellt, und es wird eine Wellenform, die im Absolutwert kleiner als der Clippingpegel ist, unterdrückt.

[0160] Der Autokorrelationsberechnungsabschnitt **212** berechnet beispielsweise die Periodizität der Eingangssprachsignalwellenform. Normalerweise wird die Tonhöhenperiode bei einer Position einer starken Spitze beobachtet. Bei dem zweiten Beispiel wird die Autokorrelationsfunktion berechnet, nachdem ein einzelner Rahmen der Eingangssprachsignalwellenform vom Mitteclipverarbeitungsabschnitt **211** mittegeclipt worden ist. Deshalb kann eine scharfe Spitze beobachtet werden.

[0161] Der Spitzedetektionsabschnitt **213** detektiert mehrere oder alle Spitzen von den vom Autokorrelationsberechnungsabschnitt **212** berechneten Autokorrelationsdaten. Kurz ausgedrückt wird der Wert r(n) des n-ten Abtastwertes der Autokorrelationsfunktion die Spitze, wenn der Wert r(n) größer als benachbarte Autokorrelationen r(n – 1) und r(n + 1) ist. Der Spitzedetektionsabschnitt **213** detektiert eine solche Spitze.

[0162] Der Anderrahmentonhöhenberechnungsabschnitt **214** berechnet eine Tonhöhe eines von dem vom Rahmenabgrenzungsabschnitt **210** abgegrenzten laufenden Rahmen verschiedenen Rahmen. Bei der vorliegenden Ausführungsform wird die Eingangssprachsignalwellenform durch den Rahmenabgrenzungsabschnitt

210 in beispielsweise einen laufenden Rahmen, einen vergangenen Rahmen und einen zukünftigen Rahmen geteilt. Bei der vorliegenden Ausführungsform wird der laufende Rahmen auf der Basis der festen Tonhöhe des vergangenen Rahmens bestimmt, und die bestimmte Tonhöhe des laufenden Rahmens wird auf der Basis der Tonhöhe des vergangenen Rahmens und der Tonhöhe des zukünftigen Rahmens fixiert. Die Idee einer präzisen Erzeugung der Tonhöhe des laufenden Rahmens aus dem vergangenen Rahmen, dem laufenden Rahmen und dem zukünftigen Rahmen wird als eine verzögerte Entscheidung bezeichnet.

[0163] Der Vergleichs/Detektions-Abschnitt 215 vergleicht die Spitzen danach, ob die vom Spitzendetektionsabschnitt **213** detektierten mehreren Spitzen in einem Tonhöhenbereich sind, der eine vorbestimmten Funktion mit der Tonhöhe des Anderrahmentonhöhenberechnungsabschnitts **214** erfüllt, und detektiert Spitzen in dem Bereich.

[0164] Der Tonhöhenentscheidungsabschnitt **216** stellt die Tonhöhe des laufenden Rahmens aus den vom Vergleichs/Detektions-Abschnitt **215** verglichenen und detektierten Spitzen fest.

[0165] Der Spitzedetektionsabschnitt **213** unter den oben beschriebenen Komponenteneinheiten und die Verarbeitung der mehreren oder aller vom Spitzendetektionsabschnitt **213** detektierten Spitzen werden unter Bezugnahme auf die **Fig.** 13 erläutert.

[0166] Die bei A in **Fig.** 13 gezeigte Eingangssprachsignalwellenform x(n) wird vom Mitteclipverarbeitungabschnitt **211** mittegeclipt, und dann wird die Wellenform r(n) der Autokorrelation wie bei B in **Fig.** 13 angedeutet vom Autokorrelationsberechnungsabschnitt **212** gefunden. Der Spitzedetektionsabschnitt **213** detektiert mehrere oder alle Spitzen, welche die Wellenform r(n) der Autokorrelation aufweist, was durch die Formel (14)

$$r(n) > r(n-1) \text{ und } r(n) > r(n+1) \tag{14}$$

ausgedrückt werden kann.

[0167] Gleichzeitig wird eine durch Normierung des Wertes der Autokorrelation r(n) erzeugte Spitze r'(n) wie bei C in **Fig.** 13 angedeutet aufgezeichnet. Die Spitze r'(n) ist die Autokorrelation r(n) dividiert durch die Autokorrelationsdaten r(0) für n = 0. Die Autokorrelationsdaten r(0), die als Spitze das Maximum ist, ist in den durch die Formel (14) ausgedrückten Spitzen nicht enthalten, da sie der Formel (14) nicht genügt. Die Spitze r'(n) wird als ein Volumen bzw. Lautstärkepegel betrachtet, das bzw. der den Grad des Seins einer Tonhöhe ausdrückt, und wird entsprechend ihrem Volumen bzw. Lautstärkepegel neu geordnet, um $r'_S(n)$, P(n) zu erzeugen. Der Wert $r'_S(n)$ ordnet r'(n) entsprechend ihrem Volumen bzw. Lautstärkepegel, wobei die folgende Bedingung

$$r'_S(0) > r'_S(1) > r'_S(2) > \ldots > r'_S(j-1) \tag{15}$$

erfüllt ist. In dieser Formel (15) stellt j die Gesamtzahl von Spitzen dar. P(n) drückt, wie bei C in **Fig.** 13 gezeigt, einen mit einer großen Spitze korrespondierenden Index aus. In der **Fig.** 13C ist der Index der größten Spitze bei einer Position n = 6 gleich P(0). Der Index der nächst größten Spitze (bei der Position n = 7) ist gleich P(1). P(n) erfüllt die Bedingung

$$r'(P(n)) = r'_S(n). \tag{16}$$

[0168] Die durch Neuordnung der normierten Funktion r'(n) der Autokorrelation r(n) erzeugte größte Spitze von $r'_S(n)$ ist gleich $r'_S(0)$. Es wird eine Tonhöhenentscheidung bzw. – feststellung in dem Fall, dass der größte oder maximale Spitzenwert $r'_S(0)$ einen durch beispielsweise k = 0,4 gegebenen vorbestimmten Wert überschreitet, erläutert.

[0169] Zuerst wird beim Überschreiten des Wertes k durch den Maximumspitzenwert $r'_S(0)$ die Tonhöhenentscheidung wie folgt ausgeführt.

[0170] Bei der vorliegenden Ausführungsform ist k auf 0,4 gesetzt. Wenn der Maximumspitzenwert $r'_S(0)$ k = 0,4 überschreitet, bedeutet dies, dass der Maximumspitzenwert $r'_S(0)$ als Maximumwert der Autokorrelation sehr hoch ist. P(0) dieses Maximumspitzenwerts $r'_S(0)$ wird vom Tonhöhenentscheidungsabschnitt **216** als die Tonhöhe des laufenden Rahmens verwendet. Infolgedessen besteht die Wahrscheinlichkeit, dass, selbst wenn ein Sprecher, der ein Ziel der Analyse sein soll, plötzlich eine Stimme wie beispielsweise „Oh!" erhebt, ein Springen der Tonhöhe nur im laufenden Rahmen realisiert werden kann, ungeachtet der Tonhöhen im vergangenen und zukünftigen Rahmen. Gleichzeitig wird festgestellt, dass die Tonhöhe zu diesem Zeitpunkt eine für den Sprecher typische Tonhöhe ist, und sie wird beibehalten. Dies ist beim Fehlen der vergangenen Tonhöhe, beispielsweise wenn die Analyse nach einem Eliminieren der Stimme des Sprechers wieder aufgenommen wird, effektiv. In diesem Fall wird P(0) wie folgt als eine typische Tonhöhe gesetzt:

$$P_t = P(0). \tag{17}$$

[0171] Wenn der Maximumspitzenwert $r'_S(0)$ kleiner als k = 0,4 ist, gilt folgendes.

[0172] Wenn die Tonhöhe P_{-1} (nachfolgend als vergangene Tonhöhe bezeichnet) des anderen Rahmens nicht von der Anderrahmentonhöhenberechnungseinheit **214** berechnet wird, das heißt, wenn die vergangene Tonhöhe P_{-1} gleich 0 ist, wird k für einen Vergleich mit dem Maximumspitzenwert $r'_S(0)$ auf 0,25 erniedrigt. Wenn der Maximumspitzenwert $r'_S(0)$ größer als k ist, wird P(0) in der Position des Maximumspitzenwertes $r'_S(0)$ vom Tonhöhenentscheidungsabschnitt **216** als die Tonhöhe des laufenden Rahmens angenommen. Zu diesem Zeitpunkt wird die Tonhöhe P(0) nicht als eine Standardtonhöhe registriert.
[0173] Andererseits wird bei Berechnung der Tonhöhe des anderen Rahmens durch den Anderrahmentonhöhenberechnungsabschnitt **214** der Maximumspitzenwert $r'_S(P_{-1})$ in einem Bereich in der Nähe der vergangenen Tonhöhe P_{-1} gesucht. In anderen Worten ausgedrückt wird die Tonhöhe des laufenden Rahmens entsprechend der Position der Spitze in einem Bereich gesucht, der eine vorbestimmte Relation mit der vergangenen Tonhöhe P_{-1} erfüllt. Insbesondere wird $r'_S(n)$ in einem Bereich von $0 < n < j$ der bereits gefundenen vergangenen Tonhöhe P_{-1} gesucht, und der

$$0{,}8P_{-1} < P(0) < 1{,}2P_{-1} \tag{18}$$

erfüllende Minimumwert wird als n_m gefunden. Je kleiner der Wert von n ist, desto größer ist die Spitze nach der Neuordnung. Die Tonhöhe $P(n_m)$ in der Position der Spitze $r'_S(n_m)$, die n_m ist, wird als ein Kandidat für die Tonhöhe des laufenden Rahmens registriert.
[0174] Ist indessen die Spitze $r'_S(n_m)$ gleich 0,3 oder größer, kann sie als die Tonhöhe angenommen werden. Wenn die Spitze $r'_S(n_m)$ kleiner als 0,3 ist, ist die Wahrscheinlichkeit, dass sie die Tonhöhe ist, niedrig, und deshalb wird $r'_S(n)$ in einem Bereich von $0 < n < j$ der schon gefundenen typischen Tonhöhe Pt gesucht, und der

$$0{,}8P_t < P(n) < 1{,}2P_t \tag{19}$$

erfüllende Minimumwert von n wird als n_r gefunden. Je kleiner der Wert von n ist, desto größer ist die Spitze nach der Neuordnung. Die Tonhöhe $P(n_r)$ in der Position der Spitze $r'_S(n_r)$, die n_r ist, wird als die Tonhöhe des laufenden Rahmens angenommen. Infolgedessen wird die Tonhöhe P_0 des laufenden Rahmens auf der Basis der Tonhöhe P_{-1} des anderen Rahmens bestimmt .
[0175] Als nächstes wird ein Verfahren zum präzisen Finden der Tonhöhe des laufenden Rahmens aus der Tonhöhe P_0 des laufenden Rahmens, der Tonhöhe P_{-1} eines einzelnen vergangenen Rahmens und der Tonhöhe P_1 eines einzelnen zukünftigen Rahmens erläutert, wobei die oben erwähnte Idee der verzögerten Entscheidung verwendet wird.
[0176] Der Grad der Tonhöhe des laufenden Rahmens wird durch den mit der Tonhöhe P_0 korrespondierenden Wert von r', das heißt $r'(P_0)$ dargestellt und wird auf R gesetzt. Die Grade der Tonhöhen des vergangenen und zukünftigen Rahmens werden auf R^- bzw. R^+ gesetzt. Demgemäss sind die Grade R, R^- und R^+ gleich $R = r'(P_0)$, $R^- = r'(P_{-1})$ bzw. $R^+ = r'(P_1)$.
[0177] Wenn der Grad R der Tonhöhe des laufenden Rahmens sowohl größer als der Grad R^- der Tonhöhe des vergangenen Rahmens als auch größer als der Grad R^+ der Tonhöhe des zukünftigen Rahmens ist, wird der Grad R der Tonhöhe des laufenden Rahmens als der höchste in der Zuverlässigkeit der Tonhöhe betrachtet. Deshalb wird die Tonhöhe P_0 des laufenden Rahmens angenommen.
[0178] Wenn der Grad R der Tonhöhe des laufenden Rahmens kleiner als der Grad R^- der Tonhöhe des vergangenen Rahmens und kleiner als der Grad R^+ der Tonhöhe des zukünftigen Rahmens ist, wobei der Grad R^- der Tonhöhe des vergangenen Rahmens größer als der Grad R^+ der Tonhöhe des zukünftigen Rahmens ist, wird $r'_S(n)$ in einem Bereich von $0 \leq n < j$ gesucht, wobei die Tonhöhe P_{-1} des zukünftigen Rahmens als der Standardton P_r, verwendet wird und der

$$0{,}8\ P_r < P(n) < 1{,}2\ P_r \tag{20}$$

erfüllende Minimumwert von n als n_a gefunden wird. Je kleiner der Wert von n ist, desto größer ist die Spitze nach der Neuordnung. Dann wird die Tonhöhe $P(n_a)$ in der Position der Spitze $r'_a(n_a)$, die gleich n_a ist, als die Tonhöhe des laufenden Rahmens angenommen.
[0179] Hierauf wird die Tonhöhenextraktionsoperation im zweiten Beispiel des Tonhöhenextraktionsverfahrens unter Bezugnahme auf das Flussdiagramm der **Fig.** 14 erläutert.
[0180] Bezüglich **Fig.** 14 wird beim Schritt 5201 zuerst eine Autokorrelationsfunktion einer Eingangssprachsignalwellenform gefunden. Insbesondere wird die Eingangssprachsignalwellenform für einen einzelnen Rahmen aus dem Rahmenabgrenzungsabschnitt 210 vom Mitteclipverarbeitungsabschnitt **211** mittegeclipt, und dann wird die Autokorrelationsfunktion der Wellenform vom Autokorrelationsberechnungsabschnitt **212** berechnet.
[0181] Beim Schritt S202 werden mehrere oder alle Spitzen (Maximumwerte), welche die Bedingungen der Formel (14) erfüllen, vom Spitzedetektionsabschnitt **213** aus der Autokorrelationsfunktion des Schrittes 5201

detektiert.

[0182] Beim Schritt 5203 werden die mehreren oder alle Spitzen, die beim Schritt 5202 detektiert werden, in der Folge ihrer Größe neu geordnet.

[0183] Beim Schritt S204 wird festgestellt, ob die Maximumspitze $r'_S(0)$ unter den beim Schritt 5203 neugeordneten Spitzen größer als 0,4 ist oder nicht. Wenn JA gewählt ist, das heißt, wenn festgestellt wird, dass die Maximumspitze $r'_S(0)$ größer als 0,4 ist, geht die Operation zum Schritt 5205 vor. Wenn andererseits NEIN gewählt ist, das heißt, wenn die Maximumspitze $r'_S(0)$ kleiner als 0,4 ist, geht die Operation zum Schritt 5206 vor.

[0184] Beim Schritt 5205 wird als Ergebnis der Entscheidung auf JA beim Schritt 5204 festgestellt, dass P(0) die Tonhöhe P_0 des laufenden Rahmens ist. P(0) wird als die typische Tonhöhe P_t gesetzt.

[0185] Beim Schritt 5206 wird bestimmt, ob die Tonhöhe P_{-1} fehlt oder nicht in einem vorhergehenden Rahmen ist. Wenn JA gewählt ist, das heißt, wenn die Tonhöhe P_{-1} fehlt, geht die Operation zum Schritt 5207 vor. Wenn andererseits NEIN gewählt ist, das heißt, wenn die Tonhöhe P_{-1} vorhanden ist, geht die Operation zum Schritt 5208 vor.

[0186] Beim Schritt 5207 wird bestimmt, ob der Maximumspitzenwert $r'_S(0)$ größer als k = 0,25 ist oder nicht. Wenn JA gewählt ist, das heißt, wenn der Maximumspitzenwert $r'_S(0)$ größer als k ist, geht die Operation zum Schritt 5208 vor. Wenn andererseits NEIN gewählt ist, das heißt, wenn der Maximumspitzenwert $r'_S(0)$ kleiner als k ist, geht die Operation zum Schritt 5209 vor.

[0187] Wenn beim Schritt S207 JA gewählt ist, das heißt, wenn der Maximumspitzenwert $r'_S(0)$ größer als k = 0,25 ist, wird beim Schritt 5208 festgestellt, dass P(0) die Tonhöhe P_0 des laufenden Rahmens ist.

[0188] Wenn beim Schritt 5207 NEIN gewählt ist, das heißt, wenn der Maximumspitzenwert $r'_S(0)$ kleiner als k = 0,25 ist, wird beim Schritt 5209 festgestellt, dass im laufenden Rahmen keine Tonhöhe vorhanden ist, das heißt P_0 = P(0) gilt.

[0189] Beim Schritt S201 wird entsprechend der Tatsache, dass die Tonhöhe P_{-1} des vergangenen Rahmens beim Schritt 5206 nicht gleich 0 ist, das heißt beim Vorhandensein der Tonhöhe festgestellt, ob der Spitzenwert bei der Tonhöhe P_{-1} des vergangenen Rahmens größer als 0,2 ist oder nicht. Wenn JA gewählt ist, das heißt, wenn die vergangene Tonhöhe P_{-1} größer als 0,2 ist, geht die Operation zum Schritt 5211 vor. Wenn NEIN gewählt ist, das heißt, wenn die vergangene Tonhöhe P_{-1} kleiner als 0,2 ist, geht die Operation zum Schritt 5214 vor.

[0190] Beim Schritt 5211 wird entsprechend der Entscheidung auf JA beim Schritt S210 der Maximumspitzenwert $r'_S(P_{-1})$ in einem Bereich von 80% bis 120% der Tonhöhe P_{-1} des vergangenen Rahmens gesucht. Kurz ausgedrückt wird $R'_S(n)$ in einem Bereich von $0 < n < j$ der bereits gefundenen vergangenen Tonhöhe P_{-1} gesucht.

[0191] Beim Schritt 5212 wird festgestellt, ob der beim Schritt 5212 gesuchte Kandidat für die Tonhöhe des laufenden Rahmens größer als der vorbestimmte Wert 0,3 ist oder nicht. Wenn JA gewählt ist, geht die Operation zum Schritt 5213 vor. Wenn NEIN gewählt ist, geht die Operation zum Schritt 5217 vor.

[0192] Beim Schritt 5213 wird entsprechend der Entscheidung auf JA beim Schritt 5212 festgestellt, dass der Kandidat für die Tonhöhe des laufenden Rahmens die Tonhöhe P_0 des laufenden Rahmens ist.

[0193] Beim Schritt 5214 wird entsprechend der Entscheidung beim Schritt 5210, dass der Spitzenwert $r'(P_{-1})$ bei der vergangenen Tonhöhe P_{-1} kleiner als 0,2 ist, festgestellt, ob der Maximumspitzenwert $r'_S(0)$ größer als 0,35 ist oder nicht. Wenn JA gewählt ist, das heißt, wenn der Maximumspitzenwert $r'_S(0)$ größer als 0,35 ist, geht die Operation zum Schritt S215 vor. Wenn NEIN gewählt ist, das heißt wenn der Maximumspitzenwert $r'_S(0)$ nicht größer als 0,35 ist, geht die Operation zum Schritt 5216 vor.

[0194] Beim Schritt 5215 wird, wenn beim Schritt 5214 JA gewählt ist, das heißt der Maximumspitzenwert $r'_S(0)$ größer als 0,35 ist, festgestellt, dass P(0) die Tonhöhe P_0 des laufenden Rahmens ist.

[0195] Beim Schritt 5216 wird, wenn beim Schritt 5214 NEIN gewählt ist, das heißt der Maximumspitzenwert $r'_S(0)$ nicht größer als 0,35 ist, festgestellt, dass im laufenden Rahmen keine Tonhöhe vorhanden ist.

[0196] Beim Schritt 5217 wird entsprechend der Entscheidung auf NEIN beim Schritt S214 der Maximumspitzenwert $r'_S(P_t)$ innerhalb eines Bereiches von 80% bis 120% der typischen Tonhöhe P_t gesucht. Kurz ausgedrückt wird $r'_S(n)$ in einem Bereich von $0 \leq n < j$ der bereits gefundenen typischen Tonhöhe P_t gesucht. Beim Schritt 5218 wird festgestellt, dass die beim Schritt 5217 gefundene Tonhöhe die Tonhöhe P_0 des laufenden Rahmens ist.

[0197] Auf diese Weise wird entsprechend dem zweiten Beispiel des Tonhöhenextraktionsverfahrens die Tonhöhe des laufenden Rahmens auf der Basis der im vergangenen Rahmen berechneten Tonhöhe festgestellt. Hierauf ist es möglich, die aus der Vergangenheit festgestellte Tonhöhe des laufenden Rahmens auf der Basis der Tonhöhe des vergangenen Rahmens, der Tonhöhe des laufenden Rahmens und der Tonhöhe des zukünftigen Rahmens präzise einzustellen.

[0198] Als nächstes wird eine Tonhöhenextraktionseinrichtung, auf die das dritte Beispiel des Tonhöhenextraktionsverfahrens angewendet ist, unter Bezugnahme auf die **Fig.** 15 erläutert. Die **Fig.** 15 ist ein funktionelles Blockschaltbild zur Erläuterung der Funktion des dritten Beispiels, wobei Darstellungen von Abschnitten, die ähnlich denen im funktionellen Blockschaltbild des zweiten Beispiels (**Fig.** 12) sind, fortgelassen sind.

[0199] Die Tonhöhenextraktionseinrichtung, auf die das dritte Beispiel des Tonhöhenextraktionsverfahrens angewendet ist, weist auf: einen Maximumspitzedetektionsabschnitt 231 zum Detektieren mehrerer oder aller Spitzen der von einem Eingangsanschluss **203** durch einen Spitzedetektionsabschnitt **213** zugeführten Autokorrelationsdaten und zum Detektieren der Maximumspitze aus den mehreren oder allen Spitzen, einen Komparator 232 zum Vergleichen des Maximumspitzenwertes aus dem Maximumspitzedetektionsabschnitt **231** und einer Schwelle eines Schwelleneinstellungsabschnitts **233**, einen Effektivtonhöhendetektionsabschnitt **235** zur Berechnung einer effektiven Tonhöhe aus über einen Eingangsanschluss **204** zugeführten Tonhöhen anderer Rahmen, und einen Multiplexer (MPX) **234**, dem die Maximumspitze aus dem Maximumspitzedetektionsabschnitt **231** und die effektive Tonhöhe aus der Effektivtonhöhendetektionseinheit **235** zugeführt sind und in welchem eine Selektion zwischen der Maximumspitze und der effektiven Tonhöhe entsprechend Ergebnissen des Vergleichs durch den Komparator **232** zur Ausgabe von „1" an einem Ausgangsanschluss **205** gesteuert werden.

[0200] Der Maximumspitzedetektionsabschnitt **231** detektiert die Maximumspitze unter den mehreren oder allen vom Spitzedetektionsabschnitt **231** detektierten Spitzen.

[0201] Der Komparator **232** vergleicht die vorbestimmte Schwelle des Schwelleneinstellungsabschnitts **233** und die Maximumspitze des Maximumspitzedetektionsabschnitts 231 im Sinne der Größe.

[0202] Der Effektivtonhöhendetektionsabschnitt **235** detektiert die effektive Tonhöhe, die in einem Tonhöhenbereich vorhanden ist, der eine vorbestimmte Relation mit der in einem von dem laufenden Rahmen verschiedenen Rahmen gefundenen Tonhöhe erfüllt.

[0203] Der MPX **234** wählt die Tonhöhe bei der Position der Maximumspitze oder die effektive Tonhöhe aus dem Effektivtonhöhendetektionsabschnitt **235** auf der Basis der Ergebnisse des Vergleichs der Schwelle und der Maximumspitze durch den Komparator **232** und gibt sie aus.

[0204] Ein konkreter Verarbeitungsfluss, der ähnlich dem des im Flussdiagramm der **Fig.** 14 des zweiten Beispiels des Tonhöhenextraktionsverfahrens ist, ist fortgelassen.

[0205] Infolgedessen wird beim dritten Beispiel des Tonhöhenextraktionsverfahrens der vorliegenden Erfindung die Maximumspitze aus mehreren oder allen Spitzen der Autokorrelation detektiert, und die Maximumspitze und die vorbestimmte Schwelle werden verglichen, wobei die Tonhöhe des laufenden Rahmens auf der Basis des Vergleichsergebnisses festgestellt wird. Gemäß diesem dritten Beispiel des Tonhöhenextraktionsverfahrens der vorliegenden Erfindung wird die Tonhöhe des laufenden Rahmens auf der Basis von in den anderen Rahmen berechneten Tonhöhen festgestellt, und die aus den Tonhöhen der anderen Rahmen festgestellte Tonhöhe des laufenden Rahmens kann auf der Basis der Tonhöhen der anderen Rahmen und der Tonhöhe des laufenden Rahmens präzise eingestellt werden.

[0206] Eine Anwendung des zweiten und dritten Beispiels des Tonhöhenextraktionsverfahrens auf den in Bezug auf die **Fig.** 1 bis 7 erläuterten MBE-Vocoder ist wie folgt. Aus den Autokorrelationsdaten des laufenden Rahmens (die für 1-Block-N-Abtastwerte-Daten gefundene Autokorrelation) werden mehrere Spitzen gefunden. Wenn die Maximumspitze unter den mehreren Spitzen gleich oder größer als eine vorbestimmte Stelle ist, wird die Position der Maximumspitze gesetzt, um eine Tonhöhenperiode zu sein. Andernfalls wird eine Spitze in einem eine vorbestimmte Relation mit einer in einem vom laufenden Rahmen verschiedenen Rahmen, beispielsweise einem vorhergehenden und/oder folgenden Rahmen gefundenen Tonhöhe erfüllenden Tonhöhenbereich gefunden. Beispielsweise wird eine Tonhöhe gefunden, die in einem ± 20%-Bereich von einer Tonhöhe eines vorhergehenden Rahmens vorhanden ist. Auf der Basis der Position dieser Spitze wird die Tonhöhe des laufenden Rahmens festgestellt. Deshalb ist es möglich, eine präzise Tonhöhe einzufangen.

[0207] Gemäß dem zweiten Beispiel des Tonhöhenextraktionsverfahrens ist es möglich, die Tonhöhe des laufenden Rahmens auf der Basis der Position der Spitze, die sich unter den mehreren aus den Autokorrelationsdaten des laufenden Rahmens des auf der rahmenweisen Basis abgegrenzten Eingangssprachsignals detektierten Spitzen befindet und die in dem die vorbestimmte Relation mit der in einem vom laufenden Rahmen verschiedenen Rahmen gefundenen Tonhöhe erfüllenden Tonhöhenbereich vorhanden ist, festzustellen. Auch ist es möglich, die Tonhöhe des laufenden Rahmens auf der Basis der Position der Spitze, die sich unter allen aus den Autokorrelationsdaten des laufenden Rahmens des auf der rahmenweisen Basis abgegrenzten Eingangssprachsignals detektierten Spitzen befindet und die in dem die vorbestimmte Relation mit der in einem vom laufenden Rahmen verschiedenen Rahmen gefundenen Tonhöhe erfüllenden Tonhöhenbereich vorhanden ist, festzustellen. Außerdem ist es wie beim dritten Beispiel möglich, die Tonhöhe des laufenden Rahmens entsprechend der Position der Maximumspitze festzustellen, wenn die Maximumspitze unter den mehreren aus den Autokorrelationsdaten des laufenden Rahmens des auf der rahmenweisen Basis abgegrenzten Eingangssprachsignals detektierten Spitzen gleich oder größer als die vorbestimmte Schwelle ist. Auch ist es möglich, die Tonhöhe des laufenden Rahmens auf der Basis der Position der in dem die vorbestimmte Relation mit der in einem vom laufenden Rahmen verschiedenen Rahmen gefundenen Tonhöhe erfüllenden Tonhöhenbereich vorhandenen Spitze festzustellen, wenn die Maximumspitze kleiner als die vorbestimmte Schwelle ist. Demgemäss wird die Wahrscheinlichkeit des Einfangens einer falschen Tonhöhe erniedrigt. Außerdem ist es selbst nach der Beseitigung der Spitze möglich, eine stabile Verfolgung in Bezug auf die in der Vergangenheit

gefundene sichere Tonhöhe auszuführen. Wenn infolgedessen mehrere Sprecher gleichzeitig sprechen, kann das Tonhöhenextraktionsverfahren auf eine Sprechertrennung zur Extraktion von Stimmen- bzw. Sprachtönen nur eines einzelnen Sprechers angewendet werden.

[0208] Indessen wird die Spektrumenveloppe von Sprachsignalen in einem einzelnen Block oder einem einzelnen Rahmen entsprechend den auf der blockweisen Basis extrahierten Tonhöhe in Bänder geteilt, wobei für jedes Band eine Stimmhaft/Stimmlos-Entscheidung ausgeführt wird. Auch im Hinblick auf die Periodizität des Spektrums wird die durch Finden der Amplitude bei jeder der Oberwellen erhaltenen Spektrumenveloppe quantisiert. Deshalb werden, wenn die Tonhöhe unsicher ist, die Stimmhaft/Stimmlos-Entscheidung und die spektrale Anpassung unsicher, wodurch die Gefahr einer Verschlechterung der Tonqualität effektiv synthetisierter Stimmen bzw. Sprachen zurückbleibt.

[0209] Kurz ausgedrückt ist es bei unklarer Tonhöhe, wenn der Versuch gemacht wird, eine wie in **Fig.** 16 durch eine gestrichelte Linie angedeutete unmögliche spektrale Anpassung in einem ersten Band zu machen, unmöglich, eine Spektrumamplitude in den folgenden Bändern zu erhalten. Selbst wenn zufällig eine spektrale Anpassung im ersten Band ausgeführt werden kann, wird das erste Band als ein stimmhaftes Band verarbeitet, wodurch abnorme Töne erzeugt werden. In der **Fig.** 16 zeigt die horizontale Achse die Frequenz und das Band an, und die vertikale Achse zeigt die Spektrumamplitude an. Die durch eine durchgezogene Linie gezeigte Wellenform zeigt die Spektrumenveloppe der Spracheingangswellenform an.

[0210] Infolgedessen wird nachfolgend ein Sprachtoncodierungsverfahren erläutert, bei welchem eine Spektrumanalyse durch Einstellen einer schmalen Bandbreite der Spektrumenveloppe ausgeführt werden kann, wenn die aus dem Eingangssprachsignal detektierte Tonhöhe unsicher ist.

[0211] Bei diesem Sprachtoncodierungsverfahren wird die Spekt rumenveloppe des Eingangssprachsignals gefunden und in mehrere Bänder geteilt. Bei dem Sprachtoncodierungsverfahren wird zur Ausführung einer Quantisierung entsprechend der Leistung jedes Bandes die Tonhöhe des Eingangssprachsignals detektiert. Bei sicherem Detektieren der Tonhöhe wird die Spektrumenveloppe in Bänder mit einer Bandbreite entsprechend der Tonhöhe geteilt, und bei einem nicht sicheren Detektieren der Tonhöhe wird die Spektrumenveloppe in Bänder mit der vorbestimmten schmaleren Bandbreite geteilt.

[0212] Bei einem sicheren Detektieren der Tonhöhe wird eine Stimmhaft/Stimmlos-Entscheidung (V/UV-Entscheidung) für jedes der durch die Teilung entsprechend der Tonhöhe erzeugten Bänder ausgeführt. Bei einem nicht sicheren Detektieren der Tonhöhe wird festgestellt, dass alle Bänder mit der vorbestimmten schmaleren Bandbreite stimmlos sind.

[0213] Gemäß diesem Sprachtoncodierungsverfahren wird, wenn die vom Eingangssprachsignal detektierte Tonhöhe sicher ist, die Spektrumenveloppe in Bänder mit der Bandbreite entsprechend der detektierten Tonhöhe geteilt, und wenn die Tonhöhe nicht sicher ist, wird die Bandbreite der Spektrumenveloppe schmal eingestellt, wodurch eine fallweise Codierung ausgeführt wird.

[0214] Ein konkretes Beispiel des Sprachcodierungsverfahrens wird nachfolgend erläutert.

[0215] Für ein solches Sprachcodierungsverfahren kann ein Codierungsverfahren zum Umwandeln von Signalen auf der blockweisen Basis in Signale auf der Frequenzachse, Teilen der Signale in mehrere Bänder und Ausführen einer V/UV-Entscheidung für jedes der Bänder angewendet werden.

[0216] Eine Verallgemeinerung dieses Codierungsverfahrens ist wie folgt: ein Sprachsignal wird in Blöcke geteilt, deren jeder eine vorbestimmte Zahl Abtastwerte, beispielsweise 256 Abtastwerte aufweist, und durch eine Orthogonaltranformation wie beispielsweise FFT in Spektrumdaten auf der Frequenzachse umgewandelt, während die Tonhöhe der Stimme bzw. Sprache in dem Block detektiert wird. Ist die Tonhöhe sicher, wird das Spektrum auf der Frequenzachse in Bänder mit einem mit der Tonhöhe korrespondierenden Intervall geteilt. Ist die detektierte Tonhöhe nicht sicher oder wird keine Tonhöhe detektiert, wird das Spektrum auf der Frequenzachse in Bänder mit einer schmaleren Bandbreite geteilt und festgestellt, dass alle Bänder stimmlos sind.

[0217] Der Codierungsfluss dieses Sprachcodierungsverfahrens wird unter Bezugnahme auf das Flussdiagramm nach **Fig.** 17 erläutert.

[0218] Bezüglich der **Fig.** 17 wird die Spektrumenveloppe des Eingangssprachsignals beim Schritt S301 gefunden. Beispielsweise ist die gefundene Spektrumenveloppe eine Wellenform (sogenanntes ursprüngliches Spektrum), die durch eine durchgezogene Linie in der **Fig.** 18 angedeutet ist.

[0219] Beim Schritt S302 wird von der beim Schritt 5301 gefundenen Spektrumenveloppe des Eingangssprachsignals eine Tonhöhe detektiert. Bei dieser Tonhöhendetektion wird zur sicheren Detektion der Tonhöhe beispielsweise ein Autokorrelationsverfahren der Mitteclipwellenform angewendet. Das Autokorrelationsverfahren der Mitteclipwellenform ist ein Verfahren zur Autokorrelationsverarbeitung einer den Clippingpegel überschreitenden Mitteclipwellenform und zum Finden der Tonhöhe.

[0220] Beim Schritt S303 wird festgestellt, ob die beim Schritt 5302 detektierte Tonhöhe sicher ist oder nicht. Beim Schritt S302 kann eine Unsicherheit wie beispielsweise ein unerwarteter Ausfall des Nehmens der Tonhöhe und eine Detektion einer Tonhöhe, die um ein ganzzahliges Vielfaches oder einen Bruch falsch ist, vorhanden sein. Solche unsicher detektierten Tonhöhen werden beim Schritt S303 unterschieden. Wenn JA gewählt ist, das heißt, wenn die detektierte Tonhöhe sicher ist, geht die Operation zum Schritt S304 vor. Ist NEIN

gewählt, das heißt, ist die detektierte Tonhöhe unsicher, geht die Operation zum Schritt S305 vor.

[0221] Beim Schritt S304 wird entsprechend der Entscheidung beim Schritt S303, dass die beim Schritt S302 detektierte Tonhöhe sicher ist, die Spektrumenveloppe in Bänder mit einer mit der sicheren Tonhöhe korrespondierenden Bandbreite geteilt. In anderen Worten ausgedrückt wird die Spektrumenveloppe auf der Frequenzachse in Bänder mit einem mit der Tonhöhe korrespondierenden Intervall geteilt.

[0222] Beim Schritt S305 wird entsprechend der Entscheidung beim Schritt S303, dass die beim Schritt S302 detektierte Tonhöhe unsicher ist, die Spektrumenveloppe in Bänder mit der schmalsten Bandbreite geteilt.

[0223] Beim Schritt S306 wird für jedes der durch die Teilung mit dem mit der Tonhöhe beim Schritt 5304 korrespondierenden Intervall erzeugten Bänder eine V/UV-Entscheidung getroffen.

[0224] Beim Schritt S307 wird festgestellt, dass alle durch die Teilung mit der schmalsten Bandbreite beim Schritt 5305 erzeugten Bänder stimmlos sind. Bei der vorliegenden Ausführungsform wird die Spektrumenveloppe wie in **Fig.** 18 gezeigt in 148 Bänder von 0 bis 147 geteilt, und diese Bänder sind obligatorisch stimmlos gemacht. Mit den 148 so geteilten sehr kleinen Bändern ist es möglich, die durch eine durchgezogene Linie gezeigte ursprüngliche Spektrumenveloppe sicher zu verfolgen bzw. nachzuvollziehen.

[0225] Beim Schritt S308 wird die Spektrumenveloppe entsprechend der Leistung jedes bei den Schritten S304 und 5305 gesetzten Bandes quantisiert. Insbesondere bei Ausführung der Teilung mit der beim Schritt 5305 ausgeführten schmalsten Bandbreite kann die Präzision der Quantisierung verbessert werden. Außerdem wird bei Verwendung eines Weißrauschens als eine Erregungsquelle für alle Bänder ein synthetisiertes Rauschen ein durch ein Spektrum der durch eine gestrichelte Linie in **Fig.** 18 gezeigten Anpassung gefärbtes Rauschen, wobei kein Gitterrauschen erzeugt wird.

[0226] Auf diese Weise wird bei dem Beispiel des Sprachcodierungsverfahrens die Bandbreite der Entscheidungsbänder der Spektrumenveloppe geändert, abhängig von der bei der Tonhöhendetektion des Eingangssprachsignals detektierten Tonhöhe. Wenn beispielsweise die Tonhöhe sicher ist, wird die Bandbreite entsprechend der Tonhöhe eingestellt, und dann wird die U/UV-Entscheidung ausgeführt. Wenn die Tonhöhe nicht sicher ist, wird die schmalste Bandbreite eingestellt (beispielsweise Teilung in 148 Bänder), wodurch alle Bänder stimmlos gemacht werden.

[0227] Wenn demgemäss die Tonhöhe unklar und unsicher ist, wird eine Spektrumanalyse eines besonderen Falles ausgeführt, wodurch keine Verschlechterung der Tonqualität der synthetisierten Stimme verursacht wird.

[0228] Bei dem wie oben beschriebenen Sprachcodierungsverfahren wird die Spektrumenveloppe mit einer mit der detektierten Tonhöhe korrespondierenden Bandbreite geteilt, wenn die aus dem Eingangssprachsignal detektierte Tonhöhe sicher ist, und die Bandbreite der Spektrumenveloppe wird verengt, wenn die Tonhöhe nicht sicher ist. Infolgedessen kann eine fallweise Codierung ausgeführt werden. Insbesondere wenn die Tonhöhe nicht klar erscheint, werden alle Bänder als stimmlose Bänder des besonderen Falles verarbeitet. Deshalb kann die Präzision der Spektrumanalyse verbessert werden, und es wird kein Rauschen erzeugt, wodurch eine Verschlechterung der Tonqualität vermieden ist.

[0229] Die Anwendung des oben beschriebenen Sprachcodierungsverfahrens auf den in Bezug auf die **Fig.** 1 bis 7 erläuterten MBE-Vocoder ist wie folgt. Für den MBE-Vocoder ist eine Tonhöhendetektion hoher Präzision erforderlich. Jedoch bei Anwendung des Sprachcodierungsverfahrens auf den MBE-Vocoder, wird, wenn die Tonhöhe nicht klar erscheint, die Teilung der Spektrumenveloppe aufs Engste bzw. Schmalste eingestellt, um alle Bänder stimmlos zu machen. Infolgedessen ist es möglich, die ursprüngliche Spektrumenveloppe exakt zu verfolgen und die Präzision der Spektrumquantisierung zu verbessern.

[0230] Indessen können bei dem Sprach-Analyse-Synthese-System wie beispielsweise dem PARCOR-Verfahren, da die Zeitsteuerung der Änderung über die Erregungsquelle auf der blockweisen Basis (rahmenweisen Basis) auf der Zeitfrequenz ist, stimmhafte und stimmlose Töne nicht zusammen in einem einzelnen Rahmen vorhanden sein. Dies hat zur Folge, dass Stimmen bzw. Sprache hoher Qualität nicht erzeugt werden können bzw. kann.

[0231] Jedoch bei der MBE-Codierung werden Stimmen bzw. Sprache in einem einzelnen Block (Rahmen) in mehrere Bänder geteilt, und für jedes der Bänder wird eine Stimmhaft/Stimmlos-Entscheidung ausgeführt, wodurch eine Verbesserung in der Tonqualität beobachtet wird. Da jedoch für jedes Band erhaltene Stimmhaft/Stimmlos-Entscheidungsdaten separat übertragen werden müssen, ist die MBE-Codierung im Sinne der Bitrate unvorteilhaft.

[0232] Im Hinblick auf den oben beschriebenen Stand der Technik wird gemäss der vorliegenden Erfindung ein hocheffizientes Codierungsverfahren, bei dem für jedes Band erhaltene Stimmhaft/Stimmlos-Entscheidungsdaten mit einer kleinen Zahl Bits ohne Verschlechterung der Tonqualität übertragen werden können, vorgeschlagen.

[0233] Das hocheffiziente Codierungsverfahren der vorliegenden Erfindung weist die Schritte auf: Finden von Daten auf der Frequenzachse durch Abgrenzung eines Eingangssprachsignals auf der Block-um-Block-Basis bzw. blockweisen Basis und umwandeln des Signals in ein Signal auf der Frequenzachse, Teilen der Daten auf der Frequenzachse in mehrere Bänder, Entscheiden für jedes der geteilten Bänder, ob das Band stimmhaft oder stimmlos ist, Detektieren eines Bandes der höchsten Frequenz der stimmhaften Bänder, und Finden von

Daten in einem Grenzpunkt zur Abgrenzung eines stimmhaften Bereichs und eines stimmlosen Bereichs auf der Frequenzachse entsprechend der Zahl Bänder von einem Band auf der niedrigeren Frequenzseite bis zum detektierten Band herauf.

[0234] Wenn das Verhältnis der Zahl stimmhafter Bänder von der niedrigeren Frequenzseite bis hoch zum detektierten Band zur Zahl stimmloser Bänder gleich oder größer als eine vorbestimmte Schwelle ist, wird die Position des detektierten Bandes als der Grenzpunkt zwischen dem stimmhaften Bereich und dem stimmlosen Bereich betrachtet. Es ist auch möglich, die Zahl Bänder auf eine vorbestimmte Zahl im Voraus zu reduzieren und infolgedessen einen einzelnen Grenzpunkt mit einer kleinen festen Zahl Bits zu übertragen.

[0235] Gemäß dem wie oben beschriebenen hocheffizienten Codierungsverfahren können, da der stimmhafte Bereich und der stimmlose Bereich in einer einzelnen Position mehrerer Bänder abgegrenzt sind, die Grenzpunktdaten mit einer kleinen Zahl Bits übertragen werden. Da auch der stimmhafte Bereich und der stimmlose Bereich für jedes Band im Block (Rahmen) festgestellt werden, kann eine synthetische Tonqualität erreicht werden.

[0236] Ein Beispiel eines solchen hocheffizienten Codierungsverfahrens wird nachfolgend erläutert.

[0237] Für das hocheffiziente Codierungsverfahren kann ein Codierungsverfahren, beispielsweise das oben erwähnte MBE-Codierungsverfahren (Multibanderregungscodierungsverfahren), bei welchem ein Signal auf der blockweisen Basis in ein Signal auf der Frequenzachse umgewandelt und dann in mehrere Bänder geteilt wird, wobei für jedes Band eine Stimmhaft/Stimmlos-Entscheidung gemacht wird, verwendet werden.

[0238] Das heißt, bei einem generell hocheffizienten Codierungsverfahren wird das Sprachsignal mit einem Intervall einer vorbestimmten Zahl Abtastwerte, beispielsweise 256 Abtastwerte, in Blöcke geteilt, und das Sprachsignal wird durch eine Orthogonaltransformation wie beispielsweise FFT in Spektrumdaten auf der Frequenzachse umgewandelt. Gleichzeitig wird die Tonhöhe der Stimme im Block extrahiert, und das Spektrum auf der Frequenzachse wird mit einem der Tonhöhe entsprechenden Intervall in Bänder geteilt und infolgedessen für jedes der geteilten Bänder eine Stimmhaft/Stimmlos-Entscheidung (V/UV-Entscheidung) ausgeführt. Die V/UV-Entscheidungsdaten werden codiert und zusammen mit Amplitudendaten übertragen.

[0239] Wenn beispielsweise das Sprach-Synthese-Analyse-System wie beispielsweise der MBE-Vocoder angenommen wird, beträgt die Abtastfrequenz f_S für das Eingangssprachsignal auf der Zeitachse normalerweise 8 kHz, und die ganze Bandbreite beträgt 3,4 kHz, wobei das effektive Band 200 bis 3400 Hz ist. Die Tonhöhennacheilung bzw. der Tonhöhenversatz von einer höheren weiblichen Stimme herunter zu einer niedrigeren männlichen Stimme oder die mit der Tonhöhenperiode korrespondierende Zahl Abtastwerte beträgt annähernd 20 bis 147. Demgemäss ändert sich die Tonhöhenfrequenz in einem Bereich von 8000/147 ≈ 54 Hz bis 8000/20 = 400 Hz. Demgemäss stehen im Bereich bis zu 3,4 kHz auf der Frequenzachse **8** bis **63** Tonhöhenimpulse oder -oberschwingungen.

[0240] Auf diese Weise wird bei in Betrachtziehung der Änderung der Zahl Bänder zwischen etwa 8 bis 63 für jedes Band aufgrund der Bandteilung mit dem mit der Tonhöhe korrespondierenden Intervall bevorzugterweise die Zahl geteilter Bänder auf eine vorbestimmte Zahl, beispielsweise 12 reduziert.

[0241] Bei dem vorliegenden Beispiel wird der Grenzpunkt zur Abgrenzung des stimmhaften Bereiches und des stimmlosen Bereiches in einer einzelnen Position aller Bänder auf der Basis der V/UV-Entscheidungsdaten für mehrere Bänder, die durch eine mit der Tonhöhe korrespondierende Teilung reduziert oder erzeugt sind, gefunden, und dann werden die Daten oder der V/UV-Code zum Anzeigen des Grenzpunktes übertragen.

[0242] Eine Detektionsoperation des Grenzpunktes zwischen dem V-Bereich und dem UV-Bereich wird unter Bezugnahme auf das Flussdiagramm nach **Fig.** 19 und eine in **Fig.** 20 gezeigte Spektrumwellenform und V/UV-Umschaltwellenform erläutert. In der folgenden Beschreibung wird die Zahl geteilter Bänder auf beispielsweise 12 reduziert angenommen. Jedoch kann eine ähnliche Detektion des Grenzpunktes auch auf den Fall der variablen Zahl Bänder, die entsprechend der ursprünglichen Tonhöhe geteilt sind, angewendet werden.

[0243] Bezüglich der **Fig.** 19 werden beim ersten Schritt S401 V/UV-Daten aller Bänder eingegeben. Wenn beispielsweise die Zahl Bänder wie in **Fig.** 20A gezeigt auf 12 vom 0-ten Band bis zum 11-ten Band reduziert wird, werden alle V/UV-Daten für alle 12 Bänder genommen.

[0244] Als nächstes wird beim Schritt 5402 festgestellt, ob es nicht mehr als einen einzigen V/UV-Umschaltpunkt oder nicht gibt. Wenn NEIN gewählt ist, das heißt, wenn es zwei oder mehr Umschaltpunkte gibt, geht die Operation zum Schritt S403 vor. Beim Schritt 5403 werden die V/UV-Daten von dem Band auf der hohen Frequenzseite abgetastet und folglich wird die Bandzahl B_{VH} der höchsten Mittenfrequenz in den V-Bändern detektiert. Bei dem Beispiel nach **Fig.** 20A werden die V/UV-Daten vom 11-ten Band auf der hohen Frequenzseite in Richtung zum 0-ten Band auf der niedrigen Frequenzseite abgetastet, und die Zahl **8** des ersten V-Bandes wird auf B_{VH} gesetzt.

[0245] Beim nächsten Schritt 5404 wird die Zahl N_V der V-Bänder durch Abtasten von 0-ten Band bis zum B_{VH}-ten Band gefunden. Bei dem Beispiel nach **Fig.** 20A gilt $N_V = 7$, da sieben Bänder vom 0-ten, 1-ten, 2-ten, 4-ten, 5-ten, 6-ten und 8-ten Band zwischen dem 0-ten Band und dem 8-ten Band V-Bänder sind.

[0246] Beim nächsten Schritt S405 wird das Verhältnis $N_V/(BVH + 1)$ der Zahl N_V der V-Bänder zur Zahl B_{VH}

+ 1 der Bänder vom 0-ten Band bis zum B_{VH}-ten Band gefunden, und es wird festgestellt, ob dieses Verhältnis gleich oder größer als eine vorbestimmte Schwelle N_{th} ist oder nicht. Bei dem Beispiel nach **Fig.** 20A gilt für das Verhältnis $N_V/(B_{VH} + 1) = 7/9 \approx 0{,}78$. Wenn die Schwelle auf beispielsweise 0,7 eingestellt ist, wird die Entscheidung JA getroffen. Wenn beim Schritt S405 JA gewählt ist, geht die Operation zum Schritt 5406 vor, wo der V/UV-Code zur Anzeige des Grenzpunktes zwischen dem V-Bereich und dem UV-Bereich so eingestellt wird, dass er B_{VH} ist. Wenn beim Schritt S405 NEIN gewählt ist, geht die Operation zum Schritt S407 vor, wo festgestellt wird, dass ein ganzzahliger Wert, beispielsweise ein Wert mit fallengelassenen Dezimalbruchstellen oder ein aufgerundeter Wert des zum Zweck der Erniedrigung des V-Grades bis herauf zum B_{VH}-Band durch Multiplikation von B_{VH} mit einer Konstanten k ($k < 1$) erzeugten Wertes $k \cdot B_{VH}$ der V/UV-Code ist. Es wird festgestellt, dass die Bänder vom 0-ten Band bis zum Band des ganzzahligen Wertes von $k \cdot B_{VH}$ V-Bänder sind, und dass Bänder auf der höheren Frequenzseite UV-Bänder sind.

[0247] Wenn andererseits beim Schritt 5402 JA gewählt ist, das heißt wenn festgestellt wird, dass es einen einzigen U/V-Umschaltpunkt oder keinen gibt, geht die Operation zum Schritt S408 vor, bei welchem festgestellt wird, ob das 0-te Band das V-Band ist oder nicht. Wen JA gewählt ist, das heißt, wenn festgestellt wird, dass das 0-te Band das V-Band ist, geht die Operation zum Schritt 5402 vor, wo ähnlich zum Schritt 5403 die Bandzahl B_{VH} für das erste V-Band von der hohen Frequenzseite gesucht wird und als der V/UV-Code gesetzt wird. Wenn beim Schritt S408 NEIN gewählt ist, das heißt, wenn festgestellt wird, dass das 0-te Band das stimmlose Band ist, geht die Operation zum Schritt 5411 vor, wo alle Bänder so eingestellt werden, dass sie UV-Bänder sind, und infolgedessen der V/UV-Code so eingestellt wird, dass er gleich 0 ist.

[0248] Das heißt, wenn es einen einzigen oder keinen V/UV-Umschaltpunkt gibt, wobei die niedrige Frequenzseite gleich V ist, wird keine Modifikation addiert. Wenn die niedrige Frequenzseite gleich UV ist, werden alle Bänder so eingestellt, dass sie UV sind.

[0249] Auf diese Weise wird das V/UV-Umschalten auf keinmal oder einmal beschränkt, und die Position in allen Bänder für das V/UV-Schalten (Umschalten und Bereichsabgrenzung) wird übertragen. Die V/UV-Codes für ein Beispiel, bei welchem die Zahl Bänder wie in **Fig.** 20A gezeigt auf 12 reduziert ist, sind folgende:

V/UV-Code	Inhalt (vom 0-ten Band bis zum 11-ten Band)		
0	0000	0000	0000
1	1000	0000	0000
2	1100	0000	0000
3	1110	0000	0000
...		...	
11	1111	1111	1110
12	1111	1111	1111,

wobei UV durch 0 und V durch 1 angezeigt ist. Es gibt 13 Typen von V/UV-Codes, die mit 4 Bits übertragen werden können. Für alle V/UV-Entscheidungskennzeichen für jedes der 12 Bänder sind 12 Bit notwendig. Jedoch kann bei den oben erwähnten V/UV-Codes das übertragene Datenvolumen für die V/UV-Entscheidung auf 4/12 = 1/3 reduziert werden.

[0250] Bei dem Beispiel nach **Fig.** 20B ist der Fall gezeigt, dass der V/UV-Code gleich 8 ist, wobei das 0-te Band bis 8-te Band so eingestellt sind, dass sie V-Bereiche sind, während das 9-te Band bis 11-te Band so eingestellt sind, dass sie UV-Bereiche sind. Indessen wird beim Schritt 5405 bei auf beispielsweise 0,8 eingestellter Schwelle N_{th}, wenn der Wert von $N_V/(B_{VH} + 1)$ wie in **Fig.** 20A gezeigt gleich $7/9 \approx 0{,}78$ ist, die Entscheidung NEIN getroffen. Deshalb wird beim Schritt 5407 der Wert von $k \cdot B_{VH}$ so eingestellt, dass er der V/UV-Code ist, wodurch die V/UV-Bereichsabgrenzung auf einer niedrigeren Frequenzseite als das 8-te Band ausgeführt wird.

[0251] Mit dem oben erwähnten Algorithmus wird das Inhaltsverhältnis der V-Bänderdeterminante der Tonqualität unter den V/UV-Daten aller ursprünglichen Bänder, beispielsweise 12 Bänder, oder in anderen Worten ausgedrückt die Änderung des V-Bandes der höchsten Mittenfrequenz, mit hoher Präzision verfolgt. Deshalb ist der Algorithmus für das Verursachen von wenig Verschlechterung der Tonqualität charakterisiert. Außerdem wird es durch Einstellen der Zahl Bänder so, dass sie so klein wie oben beschrieben sind, und Treffen der V/UV-Entscheidung für jedes Band möglich, die Bitrate bei einem Erhalt von Stimmen bzw. Sprache höherer Qualität als beim PARCOR-Verfahren zu reduzieren, wobei im Vergleich zu dem Fall des regulären MBE wenig Verschlechterung der Tonqualität verursacht wird. Insbesondere wenn die Teilungszahl auf 2 gesetzt wird und wenn ein Sprachtonmodell, bei dem die niedrige Frequenzseite stimmhaft ist und bei dem die hohe Frequenzseite stimmlos ist, vorgeschlagen wird, ist es möglich, sowohl eine signifikante Reduzierung der Bitrate und Aufrechterhaltung der Tonqualität zu erzielen.

[0252] Wie aus der obigen Beschreibung klar hervorgeht, wird das Eingangssprachsignal auf der blockweisen Basis abgegrenzt und in die Daten auf der Frequenzachse umgewandelt, so dass es in mehrere Bänder geteilt ist. Das Band der höchsten Frequenz unter den stimmhaften Bändern innerhalb jedes der geteilten Bänder wird detektiert, und es werden die Daten des Grenzpunktes zur Abgrenzung des stimmhaften Bereichs und des stimmlosen Bereichs auf der Frequenzachse entsprechend der Zahl Bänder von dem Band auf der niedrigen Frequenzseite zum detektierten Band gefunden. Deshalb ist es möglich, die Grenzpunktdaten mit einer kleinen Zahl Bits zu übertragen, während eine Verbesserung der Tonqualität erreicht wird.

[0253] Indessen werden bevorzugterweise Amplitudendaten zum Ausdrücken der Spektrumenveloppe auf der Frequenzachse parallel mit der Reduktion der Zahl Bänder auf eine vorbestimmte Zahl eingestellt. Die Umwandlung der Zahl Abtastwerte der Amplitudendaten wird unter Bezugnahme auf die **Fig.** 21 erläutert.

[0254] Bei Reduzierung der Bitrate auf beispielsweise 3 bis 4kbit/s, um die Quantisierungseffizienz weiter zu verbessern, wird bei einer skalaren Quantisierung nur das Quantisierungsrauschen erhöht, wodurch Schwierigkeiten bei der praktischen Anwendbarkeit verursacht werden. Infolgedessen wird eine Vektorquantisierung zum Sammeln mehrerer Daten in einer Gruppe oder einem Vektor, die bzw. der durch einen einzelnen Code auszudrücken ist, um die Daten ohne separate Quantisierung von bei der Codierung erhaltenen Zeitachsendaten, Frequenzachsendaten und Filterkoeffizientendaten zu quantisieren, in Betracht gezogen.

[0255] Da jedoch die Zahl von Spektrumamplitudendaten von MBE, SBE und LPC sich entsprechend der Tonhöhe ändert, ist eine Vektorquantisierung variabler Dimension erforderlich, wodurch eine Verkomplizierung der Anordnung und Schwierigkeiten beim Erhalten guter Charakteristiken verursacht werden.

[0256] Auch ist es beim Nehmen einer Interblockdifferenz (Interrahmendifferenz) von Daten vor der Quantisierung unmöglich, die Differenz zu nehmen, ohne dass die Zahlen von Daten im vorhergehenden und nachfolgenden Block (Rahmen) miteinander koinzidieren. Infolgedessen wird eine Umwandlung der Zahl von Daten guter Charakteristiken bevorzugt, obgleich es notwendig sein kann, bei der Datenverarbeitung die variable Zahl von Daten in eine vorbestimmte Zahl von Daten umzuwandeln. Im Hinblick auf den oben beschriebenen Stand der Technik wird ein Umwandlungsverfahren für die Zahl von Daten vorgeschlagen, wodurch es möglich wird, eine variable Zahl von Daten in eine vorbestimmte Zahl von Daten umzuwandeln und eine Umwandlung der Zahl von Daten guter Charakteristiken, die keine Verbindung am Anschluss- bzw. Endpunkt erzeugen, auszuführen.

[0257] Das Umwandlungsverfahren für die Datenzahl weist die Schritte auf: Nichtlineares Komprimieren von Daten, bei denen die Zahl von Wellenformdaten in einem Block oder von die Wellenform ausdrückenden Parameterdaten variabel ist, und Verwenden eines Umwandlers für die Zahl von Daten, der eine variable Zahl nichtlinearer Kompressionsdaten in eine vorbestimmte Zahl von Daten zum Vergleichen der variablen Zahl nicht linearer Kompressionsdaten auf der blockweisen Basis mit der vorbestimmten Zahl von Referenzdaten auf der blockweisen Basis in einem nichtlinearen Bereich umwandelt.

[0258] Bevorzugterweise werden Leer- bzw. Dummydaten zum Interpolieren des Wertes von den letzten Daten in einem Block bis zum ersten Block bzw, zu den ersten Daten im Block an die variable Zahl nichtlinearer Kompressionsdaten für jeden Block angehängt, um die Zahl von Daten zu erweitern und dann eine Überabtastung vom Bandbegrenzungstyp auszuführen. Die Dummydaten zum Interpolieren des Wertes von den letzten Daten in dem Block bis zu den ersten Daten in dem Block sind Daten, die keinerlei plötzliche Änderung des Wertes beim Endpunkt des Blockes mit sich bringen oder die intermittierende und diskontinuierliche Werte vermeiden. Es wird ein Typ einer Änderung im Wert, wobei der letzte Datenwert im Block in einem vorbestimmten Intervall gehalten und dann auf den ersten Datenwert im Block geändert wird und wobei der erste Datenwert in einem vorbestimmten Intervall gehalten wird, in Betracht gezogen. Bei der Überabtastung vom Bandbegrenzungstyp kann eine Orthogonaltransformation wie beispielsweise eine schnelle Fouriertransformation (FFT) und ein 0-Daten-Einsetzen in ein mit dem Mehrfachen der Überabtastung (oder Tiefpassfilterverarbeitung) korrespondieres Intervall ausgeführt werden, und dann kann eine inverse Orthogonaltransformation wie beispielsweise IFFT ausgeführt werden.

[0259] Als nichtlinear komprimierte Daten können in die Daten auf der Frequenzachse umgewandelte Audiosignale wie beispielsweise Stimmen- bzw. Sprachsignale und akustische Signale verwendet werden. Speziell können Spektrumenveloppeamplitudendaten im Fall der Multibanderregungscodierung (MBE-Codierung), Spektrumamplitudendaten und ihre Parameterdaten (LSP-Parameter α-Parameter und k-Parameter) bei Einzelbanderregungscodierung (SBE-Codierung), Oberschwingungscodierung, Subbandcodierung (SBC), Linearvorhersagecodierung (LPC), diskrete Cosinustransformation (DCT), modifizierte DCT (MDCT) oder schnelle Fouriertransformation (FFT) verwendet werden. Die in die vorbestimmte Zahl von Daten umgewandelten Daten können vektorquantisiert werden. Vor der Vektorquantisierung kann eine Interblockdifferenz der vorbestimmten Zahl von Daten für jeden Block genommen werden, und die Interblockdifferenzdaten können durch Vektorquantisierung verarbeitet werden.

[0260] Es wird möglich, die umgewandelte vorgestimmte Zahl nicht linear komprimierter Daten mit den Referenzdaten in dem nichtlinearen Bereich zu vergleichen und die Interblockdifferenz vektorzuquantisieren. Außerdem ist es möglich, die Kontinuität von Datenwerten in dem Block vor der Umwandlung der Zahl von Daten

zu erhöhen, wodurch eine Umwandlung der Zahl von Daten hoher Qualität ausgeführt wird, die keine Verbindung am Blockendpunkt erzeugt.

[0261] Ein Beispiel des oben beschriebenen Umwandlungsverfahrens für die Zahl von Daten wird unter Bezugnahme auf die Zeichnungen erläutert.

[0262] Die **Fig.** 21 zeigt eine schematische Anordnung für das Umwandlungsverfahren für die Zahl von Daten, wie es oben beschrieben ist.

[0263] Bezüglich **Fig.** 21 werden Amplitudendaten der vom MBE-Vocoder berechneten Spektrumenveloppe einem Eingangsanschluss **411** zugeführt. Wenn die Amplitude in der Position jeder Oberschwingung gefunden ist, um die Amplitudendaten, welche die wie in **Fig.** 22B gezeigte Spektrumenveloppe ausdrücken, im Hinblick auf die Periodizität des mit der durch Analysieren des wie in **Fig.** 22A gezeigten Spektrums aufweisenden Sprachsignals gefundenen Tonhöhenfrequenz ω korrespondierenden Spektrums zu finden, ändert sich die Zahl der Amplitudendaten in einem vorbestimmten effektiven Band, beispielsweise 200 bis 3400 Hz, in Abhängigkeit von der Tonhöhenfrequenz ω. Infolgedessen wird eine vorbestimmte feste Frequenz ω_C vorgeschlagen, und es werden die Amplitudendaten der Spektrumenveloppe in der Position der Oberwellen der vorbestimmten Frequenz ω_C gefunden, wodurch die Zahl von Daten konstant gemacht wird.

[0264] Bei dem Beispiel nach **Fig.** 21 wird durch einen Nichtlinearkompressionsabschnitt **412** eine variable Zahl (m_{MX} + 1) der Eingangsdaten aus dem Eingangsanschluss **411** mit einer logarithmischen Kompression in beispielsweise einen dB-Bereich komprimiert und dann durch einen Datenzahlumwandlungshauptkörper **413** in eine vorbestimmte Zahl (M) von Daten umgewandelt. Der Datenzahlumwandlungshauptkörper **413** weist einen Dummydatenanhängeabschnitt **414** und einen Bandbegrenzungstyp-Überabtastabschnitt **415** auf. Der Bandbegrenzungstyp-Überabtastabschnitt **415** ist durch einen Orthogonaltransformationsverarbeitungsabschnitt (beispielsweise FFT-Verarbeitungsabschnitt) **416**, einen O-Daten-Einnetz-Verarbeitungsabschnitt 417 und einen inversen Orthogonaltransformationsverarbeitungsabschnitt (beispielsweise IFFT-Verarbeitungsabschnitt) **418** gebildet. Mit der Bandbegrenzungstyp-Überabtastung verarbeitete Daten werden durch einen Linearinterpolationsabschnitt **419** linear interpoliert, dann durch einen Dezimierungsverarbeitungsabschnitt **420** auf eine vorbestimmte Zahl von Daten eingeschränkt und von einem Ausgangsanschluss **421** ausgegeben.

[0265] Ein Amplitudendatenarray, das aus im MBE-Vocoder berechneten (m_{MX} + 1)-Daten besteht, wird auf a(m) eingestellt. m zeigt eine nachfolgende Zahl der Oberschwingungen oder eine Bandzahl an, und m_{MX} ist der Maximumwert. Jedoch ist die Zahl von Amplitudendaten in allen Bändern einschließlich den Amplitudendaten in dem Band von m = 0 gleich (m_{MX} + 1). Die Amplitudendaten a(m) werden vom Nichtlinearkompressionsabschnitt **414** in einen dB-Bereich umgewandelt. Das heißt, mit den erzeugten Daten $a_{dB}(m)$ gilt die folgende Formel:

$$a_{dB}(m) = 20 \log_{10} a(m). \quad (21)$$

[0266] Da die Zahl (m_{MX} + 1) der mit der logarithmischen Umwandlung umgewandelten Amplitudendaten $a_{dB}(m)$ sich entsprechend der Tonhöhe ändert, werden die Amplitudendaten in die vorbestimmte Zahl (M) von Amplitudendaten $b_{dB}(m)$ umgewandelt. Diese Umwandlung ist eine Art Abtastratenumwandlung. Indessen kann die Kompressionsverarbeitung durch den nichtlinearen Kompressionsabschnitt **412** eine von der logarithmischen Kompression in den dB-Bereich verschiedene pseudologarithmische Kompressionsverarbeitung, beispielsweise ein sogenanntes μ-Gesetz oder α-Gesetz sein. Mit der Kompression der Amplitude auf diese Weise kann eine effiziente Codierung realisiert werden.

[0267] Die Abtastfrequenz f_S für das in den MBE-Vocoder eingegebene Sprachsignal auf der Frequenzachse ist normalerweise gleich 8 kHz, und die ganze Bandbreite ist 3,4 kHz mit der effektiven Bandbreite von 200 bis 3400 kHz. Der Tonhöhenversatz der mit der Tonhöhenperiode korrespondierenden Zahl Abtastwerte von einer hohen weiblichen Stimme zu einer tiefen bzw. niedrigen männlichen Stimme beträgt etwa 20 bis 147. Demgemäss wird die Tonhöhenfrequenz (Winkelfrequenz) ω in einem Bereich von 8000/147 ≈ 54 Hz bis 8000/20 = 400 Hz geändert. Deshalb stehen in einem Bereich bis zu 3,4 kHz auf der Frequenzachse etwa 8 bis 63 Tonhöhenimpulse (Oberschwingungen). Das heißt, es werden als Wellenform des dB-Bereichs auf der Frequenzachse Daten, die aus 8 bis 63 Abtastwerten bestehen, mit einer Abtastumwandlung in eine vorbestimmte Zahl Abtastwerte, beispielsweise 44 Abtastwerte, verarbeitet. Diese Abtastumwandlung korrespondiert, wie in der **Fig.** 22C gezeigt, mit dem Finden von Abtastwerden der Position der Oberschwingungen für jede vorbestimmte Tonhöhenfrequenz ω_C,.

[0268] Dann werden zur Erleichterung der FFT die (m_{MX} + 1) Kompressionsdaten $a_{dB}(m)$ durch den Dummydatenanhängeabschnitt **414** auf die Zahl N_F, beispielsweise N_F = 256 erweitert. Das heißt, mit den als Dummydaten $a'_{dB}(m)$ betrachteten Daten von (m_{MX} + 1) bis N_F werden die Kompressionsdaten unter Verwendung der folgenden Formel

$$M_{MX} + 1 \le m < N_F/2 : a'_{dB}(m) = a_{dB}(m_{MX})\ N_F/2 \le m < 3N_F/4 : a'_{dB}(m) = a_{dB}(m_{MX}) \times k_1 + a_{dB}(0) \times k_2,$$ wobei $k_1 = (3N_F/4 - n)/N_F/4)$ $k_2 = (n - N_F/2)/(N_F/4)$ gilt, $3N_F/4 \le m < N_F : a'_{dB}(m) = a_{dB}(0)$ (22)

erweitert. Wie in der **Fig.** 23 gezeigt ist, werden die ursprünglichen Amplitudendaten $a_{dB}(m)$ in einem Abschnitt von 0 bis m_{MX} platziert, und die letzten Daten $a_{dB}(m_{MX})$ in dem Block werden in einem Abschnitt $m_{MX} + 1 \leq m < N_F/2$ gehalten. Ein Abschnitt $3N_F/4 \leq m < N_F$ ist eine gefaltete Linie derart, dass die ersten Daten $a_{dB}(0)$ in dem Block gehalten sind.

[0269] Das heißt, Daten werden produziert und angefüllt, so dass ein linker und rechter Rand der ursprünglichen Wellenform zur Ratenumwandlung wie in **Fig.** 23 gezeigt graduell miteinander verbunden sind. Bei FFT ist, da die Wellenform vor der Umwandlung als eine durch eine gestrichelte Linie in **Fig.** 23 gezeigte Wiederholungswellenform betrachtet wird, der Punkt von $m = N_F$ mit $m = 0$ zu verbinden.

[0270] Wenn nach der FFT eine Filterung zur Ausführung einer Multiplikation auf der Frequenzachse auszuführen ist, wird auf der in **Fig.** 23 gezeigten ursprünglichen Achse eine Faltung ausgeführt. Deshalb wird, wenn in einem von der wie in **Fig.** 24 gezeigten ursprünglichen Wellenform verschiedenen Abschnitt ($m_{MX} < m < N_F$) einfach 0 Anfüllung ausgeführt wird, eine durch eine gestrichelte Linie R in **Fig.** 24 angedeutete Verbindung an einem diskontinuierlichen Punkt erzeugt, wodurch die normale Ratenumwandlung gestört wird. Zur Verhinderung einer solchen Unvorteilhaftigkeit werden die Dummydaten angefüllt, so dass sie wie in **Fig.** 23 gezeigt nicht solche plötzlichen Änderungen des Wertes am Blockendpunkt mit sich bringen. Neben dem konkreten Beispiel der Dummydaten wird auch in Betracht gezogen, die ganzen Daten von den letzten Daten des Blocks zu den ersten Daten des Blocks linear zu interpolieren, wie es durch eine gestrichelte Linie I in **Fig.** 23 angezeigt ist, oder gekrümmt zu interpolieren.

[0271] Als nächstes wird die zu N_F Punkten (N_F Abtastwerte) erweiterte Progression oder Datenfolge vom FFT-Verarbeitungsabschnitt **416** des Bandbegrenzungstyp-Überabtastabschnitts **415** mit einer N_F-Punkt-FFT verarbeitet, wodurch, wie in **Fig.** 25A gezeigt ein Fortschreiten bzw, eine Progression (Spektrum) von 0 bis N_F erzeugt wird. Die $(O_S - 1)N_F$-Zahl von Nullen werden in einen Raum zwischen einem Abschnitt der mit 0 bis π korrespondierenden Abschnitt der Progression und einen mit π bis 2π korrespondierenden Abschnitt durch den O-Daten-Einsetz-Verarbeitungsabschnitt **417** gefüllt. O_S ist zu diesem Zeitpunkt das Überabtastverhältnis. Beispielsweise werden im Fall von $O_S = 8$ gleich $7N_F$ Nullen in den Raum zwischen den mit 0 bis π korrespondierenden Abschnitt und den mit π bis 2π korrespondierenden Abschnitt in der Progression gefüllt, wodurch eine $8N_F$-Punktprogression erzeugt wird, beispielsweise 2048 Punkte im Fall von $N_F = 256$.

[0272] Das O-Daten-Einsetzen kann eine LPF-Verarbeitung sein. Das heißt, eine Progression von O_SN_F als die Abtastrate wird mit einer durch die fette Linie in **Fig.** 26A gezeigte Tiefpassverarbeitung mit einer Grenzfrequenz bzw. einem Abschnitt von π/8 durch eine digitale Filteroperation bei O_SN_F verarbeitet, wodurch eine Folge von Abtastwerten erzeugt wird, wie sie in **Fig.** 26B gezeigt ist. Bei dieser Filteroperation besteht die Gefahr, dass eine Verbindung, wie sie durch die gestrichelte Linie R in **Fig.** 24 gezeigt ist, erzeugt werden kann. Bei der vorliegenden Ausführungsform werden zur Vermeidung der Verbindung der linke und rechte Rand der ursprünglichen Wellenform sanft miteinander verbunden, so dass keine plötzliche Änderung im Differentialkoeffizienten verursacht wird.

[0273] Als nächstes kann bei Verarbeitung von O_SN_F Punkten, beispielsweise 2048 Punkte, mit der inversen FFT durch die IFFT-Verarbeitungseinheit **418** die in **Fig.** 27 gezeigten, mit 0_S überabgetasteten Amplitudendaten mit den Dummydaten erhalten werden. Bei Ausgabe des effektiven Abschnitts dieser Datenfolge, das heißt 0 bis $O_S \times (m_{MX} + 1)$, kann die ursprüngliche Wellenform (ursprüngliche Amplitudendaten $a_{dB}(m)$) erhalten werden, die so überabgetastet ist, dass sie ein OS-mal größere Dichte aufweisen. Dies ist eine Datenfolge, die noch von der entsprechend der Tonhöhe variablen Zahl ($m_{MX} + 1$) abhängt.

[0274] Als nächstes wird zur Umwandlung der Datenfolge in eine feste Zahl von Daten eine lineare Interpolation ausgeführt. Beispielsweise zeigt die **Fig.** 28A den Fall $m_{MX} = 19$ (wobei die Zahl aller Bänder vor der Umwandlung und die Amplitudendaten gleich 20 sind). Durch Ausführen einer 8-fachen Überabtastung mit $O_S = 8$ werden $O_S \times (m_{MX} + 1) = 160$ Abtastdaten zwischen O und π erzeugt. Die 160 Abtastdaten werden dann von der Linearinterpolationseinheit **419** in eine vorbestimmte Zahl N_M Daten, beispielsweise 2048 Daten, linear interpoliert.

[0275] Die **Fig.** 29A zeigt die von der Linearinterpolationseinheit **419** durch lineare Interpolation erzeugte vorbestimmte Zahl N, beispielsweise 2048 Daten. Um diese 2048 Abtastdaten in eine vorbestimmte Zahl von M Abtastwerten, beispielsweise 44 Abtastwerte, umzuwandeln, werden die 2048 Abtastdaten vom Einschränkungsverarbeitungsabschnitt **420** eingeschränkt. Infolgedessen werden 44-Punkt-Daten erhalten. Da es nicht notwendig ist, einen Gleichsignalwert (Gleichstromdatenwert oder den 0-ten Datenwert) zwischen dem 0-ten bis 2047-ten Abtastwert zu übertragen, können 44 Daten erzeugt werden, wobei der Wert von

nint (2048/44)·i

als der Einschränkungswert verwendet wird. Jedoch ist, da $1 \leq i \leq 44$ gilt, „nint" eine Funktion, welche die nächste ganze Zahl anzeigt.

[0276] Auf diese Weise wird die in die vorbestimmte Zahl M von Abtastwerten umgewandelte Progression $b_{dB}(n)$ erhalten, wobei $1 < n \leq M$ gilt. Es genügt, wenn notwendig, die Interblock- oder Interrahmendifferenz zu nehmen, um die Progression der festen Zahl von Daten mit der Vektorquantisierung zu verarbeiten und ihren Index zu übertragen.

[0277] Auf der Empfangsseite (Syntheseseite oder Dekodiererseite) werden aus dem Index M-Punkt-Wellenformdaten erzeugt, die eine vektorquantisierte und inversquantisierte Progression $b_{VQdB}(n)$ sind. Die Datenfolge wird durch inverse Operationen der Bandbegrenzungsüberabtastung, linearen Interpolation bzw. Einschränkung ähnlich verarbeitet und dadurch in die (m_{MX} + 1)-Punkt-Progression der notwendigen Zahl Punkte umgewandelt. Indessen kann m_{MX} (oder m_{MX} + 1) durch separat übertragene Tonhöhendaten gefunden werden. Beispielsweise kann beim Setzen der für die Abtastperiode standardisierten Tonhöhenperiode auf p die Tonhöhenfrequenz w durch 2π/p gefunden werden und als m_{MX} + 1 = inint (p/2) berechnet werden, da π/ω = p/2 ist. Die Decodierungsverarbeitung wird auf der Basis der Amplitudendaten von m_{MX} + 1 Punkten ausgeführt.

[0278] Gemäß dem oben beschriebenen Umwandlungsverfahren für die Zahl von Daten ist es, da die variable Zahl von Daten im Block nicht linear komprimiert sind und in die vorbestimmte Zahl von Daten umgewandelt werden, möglich, die Interblockdifferenz (Interrahmendifferenz) zu nehmen und die Vektorquantisierung auszuführen. Deshalb ist das Umwandlungsverfahren für die Verbesserung der Codierungseffizienz sehr effektiv. Auch werden bei der Ausführung der Bandbegrenzungstyp-Überabtastungsverarbeitung für die Datenzahlumwandlung (Abtastzahlumwandlung) die Dummydaten, beispielsweise zum Interpolieren zwischen dem letzten Datenwert im Block vor der Verarbeitung und den ersten Datenwert, addiert, um die Zahl von Daten zu erweitern. Deshalb ist es möglich, eine Unvorteilhaftigkeit wie die Erzeugung einer Verbindung am Endpunkt aufgrund der späteren Filterverarbeitung zu vermeiden und eine gute Codierung, insbesondere eine hocheffiziente Vektorquantisierung zu realisieren.

[0279] Bei Reduzierung der Bitrate auf etwa 3 bis 4 kbit/s, um die Quantisierungseffizienz weiter zu verbessern, wird das Quantisierungsrauschen bei der skalaren Quantisierung erhöht, wodurch Schwierigkeiten bei der praktischen Anwendbarkeit verursacht werden.

[0280] Infolgedessen kann die Anwendung einer Vektorquantisierung in Betracht gezogen werden. Jedoch beim Setzen der Zahl Bits des Vektorquantisierungsausgangssignals (Index) auf b erhöht sich die Größe des Codebuchs des Vektorquantisierers proportional zu 2b, und das Operationsvolumen für die Codebuchsuche erhöht sich ebenfalls proportional zu 2b. Wird jedoch die Zahl der Ausgangsbits b zu klein gemacht, wird das Quantisierungsrauschen erhöht. Deshalb werden bevorzugterweise die Größe des Codebuchs und das Operationsvolumen zum Zeitpunkt der Suche reduziert, wobei die Bitzahl b bis zu einem gewissen Grad beibehalten wird. Auch bei einer Vektorquantisierung der in die Daten auf der Frequenzachse umgewandelten Daten in diesem Zustand kann die Codierungseffizienz nicht ausreichend verbessert werden. Deshalb ist eine Technik zur weiteren Verbesserung des Kompressionsverhältnisses erforderlich.

[0281] Infolgedessen wird ein hocheffizientes Codierungsverfahren vorgeschlagen, wodurch es möglich ist, die Größe des Codebuchs des Vektorquantisierers und das Operationsvolumen zum Zeitpunkt der Suche ohne Absenkung der Zahl Ausgangsbits einer Vektorquantisierung zu reduzieren und das Kompressionsverhältnis bei der Vektorquantisierung zu verbessern.

[0282] Gemäß der vorliegenden Erfindung ist ein hocheffizientes Codierungsverfahren bereitgestellt, welches die Schritte aufweist: Teilen von Eingangsaudiosignalen in Blöcke und Umwandeln der Blocksignale in Signale auf der Frequenzachse zum Finden von Daten auf der Frequenzachse als einen Mdimensionalen Vektor, Teilen der M-dimensionalen Daten auf der Frequenzachse in mehrere Gruppen und Finden eines repräsentativen Wertes für jede der Gruppen zum Erniedrigen der M-Dimension auf eine S-Dimension, wobei S < M ist, Verarbeiten der S-dimensionalen Daten durch eine erste Vektorquantisierung, Verarbeiten von Ausgangsdaten der ersten Vektorquantisierung durch eine inverse Vektorquantisierung zum Finden eines korrespondierenden S-dimensionalen Codevektors, Expandieren des S-dimensionalen Codevektors auf einen ursprünglichen M-dimensionalen Vektor, und Verarbeiten von die Relation zwischen Daten auf der Frequenzachse des expandierten M-dimensionalen Vektors und des ursprünglichen M-dimensionalen Vektors darstellenden Daten mit einer zweiten Vektorquantisierung.

[0283] Die in Daten auf der Frequenzachse auf der blockweisen Basis umgewandelten und auf nichtlineare Weise komprimierten Daten können als die Daten auf der Frequenzachse des Mdimensionalen Vektors verwendet werden.

[0284] Gemäß einem anderen Aspekt der vorliegenden Erfindung weist das hocheffiziente Codierungsverfahren die Schritte auf: Nichtlineares Kompressimieren von durch Teilen von Eingangsaudiosignalen in Blöcke erhaltenen Daten und Umwandeln resultierender Blockdaten in Signale auf der Frequenzachse zum Finden von Daten auf der Frequenzachse als den Mdimensionalen Vektor und Verarbeiten der Daten auf der Frequenzachse des M-dimensionalen Vektors mit einer Vektorquantisierung.

[0285] Bei diesem hocheffizienten Codierungsverfahren kann die Interblock-Differenz von vektorzuquantisierenden Daten genommen und mit einer Vektorquantisierung verarbeitet werden.

[0286] Gemäß einem noch anderen Aspekt der vorliegenden Erfindung weist ein hocheffizientes Codierungsverfahren auf: Nehmen einer Interblockdifferenz von durch Teilen von Eingangsaudiosignalen auf der blockweisen Basis erhaltenen Daten und durch Umwandeln in Signale auf der Frequenzachse zum Finden von Interblockdifferenzdaten als den M-dimensionalen Vektor und Verarbeiten der Interblockdifferenzdaten des M-dimensionalen Vektors mit einer Vektorquantisierung.

[0287] Gemäß einem noch anderen Aspekt der vorliegenden Erfindung weist ein hocheffizientes Codierungsverfahren die Schritte auf: Teilen von Eingangsaudiosignalen in Blöcke und Umwandeln der Blocksignale in Signale auf der Frequenzachse zum Umwandeln einer Amplitude des Spektrums in eine dB-Bereichsamplitude, um so Daten auf der Frequenzachse als einen M-dimensionalen Vektor zu finden, Teilen der M-dimensionalen Daten auf der Frequenzachse in mehrere Gruppen und Finden von Mittelwerten für die Gruppen zum Erniedrigen der M-Dimension auf eine S-Dimension, wobei S < M gilt, Verarbeiten von S-dimensionalen Mittelwertdaten mit einer ersten Vektorquantisierung, Verarbeiten von Ausgangsdaten der ersten Vektorquantisierung mit einer inversen Vektorquantisierung zum Finden eines korrespondierenden S-dimensionalen Codevektors, Expandieren des S-dimensionalen Codevektors auf einen ursprünglichen M-dimensionalen Vektor, und Verarbeiten von Differenzdaten zwischen Daten auf der Frequenzachse des expandierten M-dimensionalen Vektors und des ursprünglichen Mdimensionalen Vektors mit einer zweiten Vektorquantisierung.

[0288] Bei einem solchen hocheffizienten Codierungsverfahren wird es durch Vektorquantisierung mit einem hierarchischen Codebuch zur Erniedrigung der M-Dimension auf die S-Dimension und Ausführung der Vektorquantisierung, wobei S < M gilt, möglich, das Operationsvolumen der Codebuchsuche oder die Codebuchgröße zu verkleinern. Infolgedessen wird es möglich, einen effektiven Gebrauch vom Fehlerkorrekturcode zu machen. Andererseits kann die Quantisierungsqualität durch Ausführen der Vektorquantisierung nach einer nichtlinearen Kompression von Daten auf der Frequenzachse- verbessert werden, während die Kompressionseffizienz durch Nehmen der Interblockdifferenz weiter verbessert werden kann.

[0289] Eine bevorzugte Ausführungsform des oben beschriebenen hocheffizienten Codierungsverfahrens wird unter Bezugnahme auf die Zeichnungen erläutert.

[0290] Die **Fig.** 30 zeigt eine schematische Anordnung eines Codierers zur Erläuterung des hocheffizienten Codierungsverfahrens gemäß einer Ausführungsform der vorliegenden Erfindung.

[0291] Bei der **Fig.** 30 werden einem Eingangsanschluss **611** Sprachsignale oder akustische Signale zugeführt, um von einem Frequenzachsentransformationsprozessor **612** in Spektrumamplitudendaten auf der Frequenzachse umgewandelt zu werden. Der Frequenzachsentransformationsprozessor **12** weist auf: einen Blockbildungsabschnitt **612a** zum Teilen von Eingangssignalen auf der Frequenzachse in Blöcke, deren jeder aus einer vorbestimmten Zahl von Abtastwerten, hier n Abtastwerte, besteht, einen Orthogonaltransformationsabschnitt **612b** für beispielsweise eine schnelle Fouriertransformation (FFT), und einen Datenprozessor **612c** zum Finden der für Eigenschaften einer Spektrumenveloppe repräsentativen Amplitudeninformation. Ein Ausgangssignal aus dem Frequenzachsentransformationsprozessor **612** wird über einen fakultativen nichtlinearen Kompressionsabschnitt **613** zur Umwandlung in dB-Bereichsdaten und einen Prozessor 614 zum Nehmen der Interblockdifferenz ein Vektorquantisierer **615** zugeführt. Im Vektorquantisierer **615** wird eine vorbestimmte Zahl Abtastwerte, hier M Abtastwerte, genommen und in einen M-dimensionalen Vektor quantisiert und mit einer Vektorquantisierung verarbeitet. Generell ist die M-dimensionale Vektorquantisierung eine Operation der Suche nach einem Codevektor, der den kürzesten Abstand im Mdimensionalen Raum zum dimensionalen Eingangsvektor aus einem Codebuch aufweist, um einen Index des gesuchten Codevektors aus einem Ausgangsanschluss **616** auszugeben. Der Vektorquantisierer **615** der in **Fig.** 30 gezeigten Ausführungsform weist eine hierarchische Struktur derart auf, dass eine zweistufige Vektorquantisierung am Eingangsvektor ausgeführt wird.

[0292] Das heißt, bei dem in **Fig.** 30 gezeigten Vektorquantisierer **615** werden Daten des M-dimensionalen Vektors (Daten auf der Frequenzachse) als eine Einheit für die Vektorquantisierung zu einem Dimensionsverkleinerungsabschnitt **621** übertragen, in welchem die Daten in mehrere Gruppen geteilt werden und ein repräsentativer Wert in jeder Gruppe zur Verkleinerung der Zahl der Dimension auf S, wobei S < M ist, gefunden wird. Die **Fig.** 31 zeigt ein konkretes Beispiel von Elementen eines in den Vektorquantisierer **615** eingegebenen Mdimensionalen Vektors X, das heißt M Einheiten von Amplitudendaten x(n) auf der Frequenzachse, wobei $1 \leq n \leq M$ ist. Diese M Einheiten der Amplitudendaten x(n) werden in beispielsweise vier Abtastwerte gruppiert, und für jeden dieser vier Abtastwerte wird ein repräsentativer Wert, beispielsweise ein Mittelwert y_i gefunden. Hierauf ergibt sich ein S-dimensionaler Vektor Y, der, wie in **Fig.** 32 gezeigt, aus S Einheiten der Mittelwertdaten y_1 bis y_S besteht, wobei S = M/4 gilt.

[0293] Diese S-dimensionalen Vektordaten werden durch einen Sdimensionalen Vektorquantisierer **622** mit einer Vektorquantisierung verarbeitet. Das heißt, es wird der dem S-dimensionalen Eingangscodevektor im S-dimensionalen Raum nächstliegende Codevektor unter den S-dimensionalen Codevektoren im Codebuch des S-dimensionalen Vektorquantisierers **622** gesucht. Indexdaten des so gesuchten Codevektors werden an einem Ausgangsanschluss **626** ausgegeben. Der so gesuchte Codevektor, das heißt der durch inverse Vektorquantisierung des Ausgangsvektors erhaltene Codevektor wird zu einem Dimensionsexpandierungsabschnitt **623** übertragen. Die **Fig.** 33 zeigt Elemente Y_{VQI} bis y_{VQS} des S-dimensionalen Vektors Y_{VQ} als ein lokales Dekodiererausgangssignal, das durch Vektorquantisierung und dann inverse Quantisierung des aus S-Einheiten von in **Fig.** 32 gezeigten Mittelwertdaten y_1 bis y_S bestehenden S-dimensionalen Vektors Y, in anderen Worten ausgedrückt durch Ausgeben des durch Quantisierung durch das Codebuch des Vektorquantisierers **622** gesuchten Codevektors erhalten wird.

[0294] Der Dimensionsexpandierungsabschnitt **623** expandiert den oben erwähnten S-dimensionalen Codevektor auf einen ursprünglichen M-dimensionalen Vektor. Die **Fig.** 34 zeigt ein Beispiel der Elemente des expandierten M-dimensionalen Vektors. Aus der **Fig.** 34 ist klar zu entnehmen, dass der aus 4S = M Elementen bestehende M-dimensionale Vektor durch Erhöhung der Elemente y_{VQ1} bis y_{VQS} des invers vektorquantisieruen S-dimensionalen Vektors Y_{VQ} erhalten wird. Die zweite Vektorquantisierung wird an Daten ausgeführt, welche die Relation zwischen dem expandierten M-dimensionalen Vektor und den Daten auf der Frequenzachse des ursprünglichen M-dimensionalen Vektors anzeigen.

[0295] Bei der Ausführungsform nach **Fig.** 30 werden die expandierten M-dimensionalen Vektordaten aus dem Dimensionsexpandierungsabschnitt **623** zu einem Subtrahierer 624 zum Subtrahieren des ursprünglichen M-dimensionalen Vektors von den Daten auf der Frequenzachse übertragen, wodurch S Einheiten von Vektordaten erzeugt werden, welche die Relation zwischen dem von der S-Dimension expandierten M-dimensionalen Vektor und dem ursprünglichen M-dimensionalen Vektor anzeigen. Die **Fig.** 35 zeigt M Einheiten aus Daten r_1 bis r_m die bei der Subtraktion der Elemente des in **Fig.** 34 gezeigten expandierten Mdimensionalen Vektors von den M Einheiten von Amplitudendaten x(n) auf der Frequenzachse, die jeweilige Elemente des in **Fig.** 31 gezeigten M-dimensionalen Vektors X sind, erhalten werden. Vier Abtastwerte jeder dieser M Einheiten aus Daten r_1 bis r_M werden als Sätze oder Vektoren gruppiert, um 5 Einheiten der vier dimensionalen Vektoren R_1 bis R_S zu erzeugen.

[0296] Die aus dem Subtrahierer **624** erhaltenen S Einheiten aus Vektoren werden durch S Einheiten aus Vektorquantisierern **625$_1$** bis **625$_S$** einer Vektorquantisierergruppe **625** mit einer Vektorquantisierung verarbeitet. Ein aus jedem der Vektorquantisierer **6251** bis **6255** ausgegebener Index wird von Ausgangsanschlüssen **6271** bis **6275** ausgegeben. Die **Fig.** 36 zeigt Elemente r_{VQ1} bis r_{VQ4}, r_{VQ5} bis r_{VQ8}, ... r_{VQM} der jeweiligen vierdimensionalen Vektoren R_{VQ1} bis R_{VQS}, die aus der Vektorquantisierung der in **Fig.** 35 gezeigten vierdimensionalen Vektoren R_1 bis R_S resultieren, wobei die Vektorquantisierer **625$_1$** bis **625$_5$** als die jeweiligen vierdimensionalen Vektorquantisierer verwendet werden.

[0297] Durch die oben beschriebene hierarchische zweistufige Vektorquantisierung wird es möglich, das Operationsvolumen für Codebuchsuche und den Speicherraum für das Codebuch wie beispielsweise die ROM-Kapazität zu verkleinern. Auch wird es möglich, eine effektive Anwendung der Fehlerkorrekturcodes durch vorzugsweise Fehlerkorrekturcodierung für die aus dem Ausgangsanschluss **626** erhaltenen Indizes der oberen Ordnung durchzuführen. Indessen ist die hierarchische Struktur des Vektorquantisierers **615** nicht auf zwei Stufen beschränkt, sondern kann auch drei oder mehr Stufen einer Vektorquantisierung aufweisen.

[0298] Indessen müssen die jeweiligen Komponenten der **Fig.** 30 nicht als Hardware ausgebildet sein, sondern können unter Verwendung eines sogenannten digitalen Signalprozessors (DSP) durch Softwaretechniken ausgeführt werden. Der Vektorquantisierer **615** enthält einen Addierer 628 zum Summieren der Elemente der quantisierten Daten aus dem ersten und zweiten Vektorquantisierer 622, 625, um M Einheiten der quantisierten Daten zu erzeugen. Das heißt, die M Einheiten der expandierten M-dimensionalen Daten aus dem Dimensionsexpandierungsabschnitt **623** werden zu den M Einheiten der Elementdaten jeder der S Einheiten der Codevektoren aus den Vektorquantisierern **625$_1$**, **625$_S$** addiert, um M Einheiten aus Daten aus einem Ausgangsanschluss **629** auszugeben. Der Addierer **628** wird zum Nehmen einer später erläuterten Interblock- oder Interrahmendifferenz verwendet und kann in dem Fall, dass eine solche Interblockdifferenz nicht genommen wird, fortgelassen sein.

[0299] Die **Fig.** 37 zeigt eine schematische Anordnung eines Codierers zur Illustration des hocheffizienten Codierungsverfahrens als eine zweite Ausführungsform der vorliegenden Erfindung.

[0300] Bei der **Fig.** 37 werden Audiosignale, beispielsweise Sprachsignale oder akustische Signale, einem Eingangsanschluss **611** zugeführt, werden von einem Frequenzachsentransformationsprozessors **612** in Blöcke geteilt, deren jeder aus N Einheiten von Abtastwerten besteht, und die erzeugten Daten werden zu einem Nichtlinearkompressionsabschnitt **613** übertragen, bei dem eine nichtlineare Kompression zur Umwandlung der Daten in beispielsweise dB-Bereichsdaten ausgeführt wird. M Einheiten der erzeugten nichtlinear komprimierten Daten werden zu einem M-dimensionalen Vektor gesammelt, der dann von einem Vektorquantisierer **615** mit einer Vektorquantisierung verarbeitet und dann von einem Ausgangsanschluss **616** ausgegeben wird. Der Vektorquantisierer **615** kann eine hierarchische Struktur aus zwei Stufen oder drei Stufen oder mehr Stufen aufweisen oder kann so ausgebildet sein, dass er eine gewöhnliche Einstufenvektorquantisierung ausführt, ohne dass er eine hierarchische Struktur aufweist. Der Nichtlinearkompressionsabschnitt **613** kann so ausgebildet sein, dass er eine sogenannte pseudologarithmische μ-Gesetz- oder A-Gesetz-Kompression anstelle einer log-Kompression (logarithmische Kompression) der Umwandlung der Daten in dB-Bereichsdaten ausführt. Infolgedessen kann durch eine logarithmische Amplitudentransformation, Kompression und lineare Codierung eine effiziente Codierung realisiert werden.

[0301] Die **Fig.** 38 zeigt eine schematische Anordnung eines Codierers zur Erläuterung des hocheffizienten Codierungsverfahrens als eine dritte Ausführungsform der Erfindung.

[0302] Bei der **Fig.** 38 werden einem Eingangsanschluss zugeführte Audiosignale von einem Frequenzachsentransformationsprozessor **612** in blockweise Daten geteilt und werden in Daten auf der Frequenzach-

se geändert. Die resultierenden Daten werden über einen fakultativen Nichtlinearkompressionsabschnitt **613** zu einem Prozessor 614 zum Nehmen der Interblockdifferenz übertragen. Indessen wird, wenn die Blöcke der N Einheiten aus Abtastwerten teilweise mit benachbarten Blöcken überlappt sind und auf der Zeitachse auf der rahmenweise Basis angeordnet sind, wobei jeder Rahmen aus L Einheiten aus Abtastwerten besteht, wobei L < N ist, vom Prozessor **612** eine Rahmendifferenz genommen. Die M Einheiten von Daten, bei denen die Interblockdifferenz oder die Interrahmendifferenz genommen worden ist, werden zu einem M-dimensionalen Vektorquantisierer **615** übertragen. Die vom M-dimensionalen Vektorquantisierer **615** quantisierten Indexdaten werden von einem Ausgangsanschluss **616** ausgegeben. Der Vektorquantisierer **615** kann eine Multischichtstruktur aufweisen oder auch nicht.

[0303] Der Prozessor 614 zum Nehmen der Interblock- oder Interrahmendifferenz kann so ausgebildet sein, dass er Eingangsdaten um einen einzelnen Block oder einzelnen Rahmen verzögert, um die Differenz von den ursprünglichen Daten zu nehmen, die nicht verzögert sind. Jedoch beim Beispiel der **Fig.** 38 ist ein Subtrahierer **631** mit einer Eingangsseite des Vektorquantisierers **615** verbunden. Ein aus M Einheiten von Elementdaten bestehender Codevektor aus dem M-dimensionalen Vektorquantisierer **615** wird um einen einzelnen Block oder Rahmen verzögert und von den Eingangsdaten (M-dimensionaler Vektor) subtrahiert. Da in diesem Fall die Differenzdaten der vektorquantisierten Daten genommen wird, wird der Codevektor aus dem Vektorquantisierer **615** zu einem Addierer **632** übertragen. Ein Ausgangssignal aus dem Addierer **632** wird von einer Blockverzögerungs- oder Rahmenverzögerungsschaltung **633** verzögert und von einem Multiplizierer **634** mit einem Koeffizienten α multipliziert, das dann zum Addierer **632** übertragen wird. Ein Ausgangssignal aus dem Multiplizierer **634** wird zum Subtrahierer **631** übertragen. Indessen werden bei Verwendung der in **Fig.** 30 gezeigten zweistufigen hierarchischen Struktur beim Mdimensionalen Vektorquantisierer **615** die Daten aus einem Ausgangsanschluss 629 zum Addierer **632** als ein M-dimensionaler Codevektor zur Vektorquantisierung übertragen.

[0304] Durch Nehmen der Interblock- oder Interrahmendifferenz kann ein Präsenzbereich der Eingangsamplitudendaten auf der Frequenzachse im M-dimensionalen Raum enger bzw. schmaler gemacht werden. Dies deshalb, weil die Amplitudenänderungen des Spektrums gewöhnlich klein sind und eine starke Korrelation zwischen den Block- oder Rahmenintervallen zeigen. Folglich kann das Quantisierungsrauschen reduziert werden und infolgedessen kann die Datenkompressionseffizienz weiter verbessert werden.

[0305] Als nächstes wird nachfolgend eine konkrete Ausführungsform der vorliegenden Erfindung erläutert, bei der die spektralen Amplitudendaten von durch einen Frequenzachsentransformationsprozessor **612** erhaltenen Daten auf der Frequenzachse von einem Nichtlinearkompressionsabschnitt **613** in Amplitudendaten in einem dB-Bereich umgewandelt werden, um eine wie in **Fig.** 38 gezeigte Interblock- oder Interrahmendifferenz zu finden, und bei der die resultierenden Daten von einem Multischichtvektorquantisierer **615** mit einer wie in **Fig.** 30 gezeigten M-dimensionalen Vektorquantisierung verarbeitet werden. Obgleich bei dem Frequenzachsentransformationsprozessors **612** eine Vielfalt von Codierungssystemen angenommen werden kann, kann, wie später erläutert, eine analytische Multibanderregungs-Verarbeitung (analytische MBE-Verarbeitung) angewendet werden. Bei der Blockbildung durch den Frequenzachsentransformationsprozessor **612** werden die N Abtastblockdaten auf der Zeitachse auf der blockweisen Basis angeordnet, wobei jeder Rahmen aus L Einheiten von Abtastwerten besteht. Die Analyse wird für einen Block, der aus N Einheiten von Abtastwerten besteht, ausgeführt, und die Ergebnisse der Analyse werden in einem Intervall aus L Einheiten von Abtastwerten für jeden Rahmen erhalten (oder aktualisiert).

[0306] Es sei angenommen, dass der Wert von Daten wie beispielsweise Daten für die Spektrumamplitude als die aus dem Frequenzachsentransformationsprozessor **612** erhaltenen Ergebnisse der MBE-Analyse gleich a(m) ist, und dass eine Zahl von (m_{MX} + 1) von Abtastwerten für jeden Rahmen erhalten wird, wobei $0 \leq m \leq m_{MX}$ gilt .

[0307] Wenn durch Umwandlung der Zahl (m_{MX} + 1) von Abtastwerten aus Amplitudenwerten a(m) in dB-Bereichswerte erhaltene Daten gleich a_{dB} sind, gilt ähnlich wie bei der oben erwähnten Formel (21)

$$a_{dB}(m) = 20\log_{10}a(m). \quad (23)$$

[0308] Bei der MBE-Analyse wird die Zahl (m_{MX} + 1) von Abtastwerten für jeden Rahmen abhängig von der Tonhöhenperiode geändert. Für die Interrahmendifferenz und die Vektorquantisierung ist es wünschenswert, die Zahl der in jedem Rahmen oder Block vorhandenen dB-Rmplitudenwerte $a_{dB}(m)$ konstant zu halten. Aus diesem Grund wird die (m_{MX} + 1)-Zahl der dB-Amplitudenwerte $a_{dB}(m)$ in eine konstante Zahl M von Daten $b_{dB}(n)$ umgewandelt. Die Zahl n von Abtastpunkten ist so gewählt, dass sie für jeden Rahmen oder Block einen Wert $1 \leq n \leq M$ annimmt. Die Daten für n = 0, die mit dem dB-Amplitudenwert $a_{dB}(0)$ für m = 0 korrespondieren, weisen eine mit der Gleichsignalkomponente korrespondierende Amplitude auf und werden folglich nicht übertragen. Das heißt sie sind ständig auf 0 gesetzt.

[0309] Durch Nehmen der Interrahmendifferenz nach der Umwandlung in dB-Bereichsdaten wird es möglich, den Bereich der Präsenz der oben erwähnten Daten $b_{dB}(n)$ einzuengen. Dies deshalb, weil die Spektrumamp-

litude nur bei seltenen Gelegenheiten im Lauf eines Rahmenintervalls, beispielsweise etwa 20 ms, signifikant geändert wird und folglich eine starke Korrelation zeigt. Das heißt, die Vektorquantisierung wird an dem folgenden Wert $c_{dB}(n)$,

$$c_{dB}(n) = b_{dB}(n) - b'_{dB}(n), \tag{24}$$

von dem die Differenz genommen worden ist, ausgeführt. In dieser Formel ist b'dg(n) ein vorhergesagter Wert von $b_{dB}(n)$, und bedeutet

$$b'_{dB}(n) = \alpha \cdot b''_{dB}(n)p, \tag{25}$$

was durch Multiplizieren eines Ausgangssignals $b''_{dB}(n)p$ mit einem Koeffizienten α durch einen Multiplizierer **634** erhalten wird, wobei $b''_{dB}(n)p$ durch Verzögerung des invers quantisierten Ausgangssignals $b''_{dB}(n)$ aus dem Vektorquantisierer **615** (zum oben erwähnten Codevektor äquivalentes lokales Dekodiererausgangssignal) durch eine Verzögerungsschaltung 633 um einen einzelnen Rahmen erhalten wird, wobei p den Zustand anzeigt, dass er der vorhergehenden Rahmen ist.

[0310] Wenn die Interrahmenamplitudendifferenz auf diese Weise genommen wird, treten, obgleich das Quantisierungsrauschen weiter reduziert werden kann, leichter Codefehler auf. Dies deshalb, weil ein Fehler in einem gegebenen Rahmen auf sukzessive sich anschließende Rahmen fortgepflanzt wird. Folglich wird α auf etwa 0,7 bis 0,8 gesetzt, um eine sogenannte Leckdifferenz zu nehmen. Wenn das System stärker gegen Codefehler sein soll, ist es möglich, α sogar auf 0 zu reduzieren, das heißt, die Interrahmendifferenzen nicht zu nehmen, um zum nächsten Verarbeitungsschritt vorzugehen. In einem solchen Fall ist es notwendig, eine balancierte Leistung des ganzen Systems zu berücksichtigen.

[0311] Eine Ausführungsform, bei der die Interrahmendifferenzdaten $c_{dB}(n)$ quantisiert werden, das heißt bei welcher ein Array $c_{dB}(n)$ als der M Einheiten von Elementen aufweisende M-dimensionale Vektor vektorquantisiert wird, wird nachfolgend erläutert. Es kann auch der Fall, dass die Differenz nicht genommen wird, in $c_{dB}(n)$ enthalten sein, wenn α = 0 in Betracht gezogen wird. Die M Einheiten von Daten, die M-dimensional vektorzuquantisieren sind, werden durch x(n) ersetzt. Bei der vorliegenden Ausführungsform gilt $x(n) = c_{dB}(n)$ und $1 \leq N \leq M$. Mit der Zahl b von Bits des Index des M-dimensionalen Vektorquantisierungsausgangssignals ist es logisch möglich, eine gerade Vektorquantisierung einer direkten Suche eines Codebuchs, das eine Zahl von M-Dimension × 2^b Codevektoren aufweist, auszuführen. Jedoch nimmt das Operationsvolumen der Codebuchsuche bei der Vektorquantisierung proportional zu $M2^b$ zu und ebenso die Tabellen-ROM-Größe. Es ist deshalb praktischer, eine Vektorquantisierung zu verwenden, die ein strukturiertes Codebuch aufweist. Bei der vorliegenden Ausführungsform ist der M-dimensionale Vektor in mehrere niedrig dimensionale Vektoren geteilt, und es wird ein Mittelwert jedes der niedrig dimensionalen Vektoren berechnet. Die niedrig dimensionalen Vektoren werden in Vektoren geteilt, die aus diesen Mittelwerten (obere Ordnungsschicht) und von den Mittelwerten befreiten Vektoren (niedrige Ordnungsschichten) bestehen, von denen jeder dann mit einer Vektorquantisierung verarbeitet wird.

[0312] Die M Einheiten von Daten x(n), beispielsweise die Differenzdaten $c_{dB}(n)$, werden in S Einheiten von Vektoren geteilt:

$$\begin{aligned}
X_1 &= (x(1),\ x(2),\ldots,\ x(d_1))^t \\
X_2 &= (x(d_1+1),\ x(d_1+2),\ldots,\ x(d_1+d_2))^t \\
&\vdots \\
X_S &= (x(d_1+d_2+\ldots d_{S-1}+1),\ x(d_1+d_2+\ldots+d_{S-1}+2),\ldots, \\
&\quad x(d_1+d_2+\ldots+d_S))^t.
\end{aligned} \tag{26}$$

[0313] In der obigen Formel (26) drücken X_1, X_2, ..., X_S Vektoren der Dimensionen d_1, d_1, ... bzw. d_S aus, wobei $d_1 + d_2 + \ldots + d_S = M$ ist. t bezeichnet eine Vektortransposition. Das zuvor erwähnte und in **Fig.** 31 gezeigte konkrete Beispiel korrespondiert mit dem Fall, bei welchem die Dimensionen sämtlicher Vektoren X_1, X_2, ..., X_S alle auf 4 gesetzt sind, das heißt $d_1 = d_2 = \ldots = d_S = 4$ gilt.

[0314] Wenn Mittelwerte der Elemente der S Einheiten von Vektoren X_1, X_2, ..., X_S gleich y_1, y_2, ... bzw. y_S sind, kann $y_i (1 \leq i \leq S)$ durch

$$y_i = \frac{1}{d_i} \sum_{k=1}^{d_i} x(g_i + k) \tag{27}$$

ausgedrückt werden, wobei

$$g_i = \sum_{k=1}^{i-1} d_i \qquad (i > 1)$$

$g_i = 0$ (i = 1)

gilt. Die S-dimensionalen Mittelwerte, die diese Mittelwerte als Elemente haben, sind durch die Formel (28)

$$Y = (Y_1, Y_2, ..., Y_S)^t \tag{28}$$

definiert. Dies korrespondiert mit der **Fig.** 32. Dieser S-dimensionale Vektor Y wird zuerst vektorquantisiert. Während eine Vielfalt der Methoden zur Vektorquantisierung des Vektors Y in Betracht gezogen werden kann, beispielsweise gerade Vektorquantisierung, usw., wird bei der vorliegenden Ausführungsform die Form-Verstärkungs-Vektorquantisierung angewendet. Die Form-Verstärkungs-Vektorquantisierung ist in M. J. Sabin, R. M. Gray, „Product Code Vector Quantizer for Waveform and Voice Coding", IEEE Trans. On ASSP, Vol. ASSP-32, Nr. 3, Juni 1984 beschrieben.

[0315] Es sei angenommen, dass das Ergebnis des vektorquantisierten S-dimensionalen Vektors Y gleich YVQ ist, was durch die Formel (29)

$$Y_{VQ} = (Y_{VQ1}, Y_{VQ2}, ..., y_{VQS}) \tag{29}$$

ausgedrückt werden kann. Y_{VQ} kann als eine schematische Form oder ein charakteristisches Volumen des ursprünglichen Arrays x(n) (≡ $c_{dB}(n)$, 1 ≤ n ≤ M) betrachtet werden. Demgemäss benötigt er einen relativ starken Schutz gegen Übertragungsfehler.

[0316] Dann wird auf der Basis des S-dimensionalen Vektors Y_{VQ} das Eingangsarray x(n) des ursprünglichen M-dimensionalen Vektors (= $c_{dB}(n)$) angenommen oder auf die eine oder andere Weise dimensionsexpandiert. Ein Fehlersignal zwischen dem angenommenen Wert und dem ursprünglichen Eingangsarray hat ein Eingangssignal zur Vektorquantisierung auf der nächsten Stufe zu sein. Als typische Verfahren zur Annahme gibt es eine nichtlineare Interpolation, wie sie in A. Gersho, „Optimal Non-linear Interpolative Vector Quantization", IEEE Trans. On Comm. Vol. 38, Nr. 9, Sept. 1990 beschrieben ist, Splineinterpolation, Multiterminterpolation, gerade Interpolation (Interpolation erster Ordnung), Halten O-ter Ordnung usw. Wenn bei dieser Stufe eine exzellente Interpolation ausgeführt wird, wird der Präsenzbereich des Eingangsvektors für die nächststufige Vektorquantisierunq enger bzw. schmaler gemacht, wodurch eine Quantisierung mit weniger Störung ermöglicht ist. Bei der vorliegenden Ausführungsform wird das in **Fig.** 34 gezeigte einfachste Halten O-ter Ordnung angewendet.

[0317] Wenn die mittelwertfreien Vektoren, die mit S Einheiten von Vektoren korrespondieren, das heißt die von vorquantisierten Mittelwerten befreiten restlichen Vektoren mit $R_1, R_2, ..., R_S$ bezeichnet werden, werden diese Vektoren $R_1, R_2, ..., R_S$ durch die folgende Formel

$$\begin{aligned} R_1 &= X_1 - y_{VQ1} I_1 & x(1) - y_{VQ1} \\ &= & x(2) - y_{VQ1} \\ \vdots & \vdots & \vdots \\ R_S &= X_S - y_{VQS} I_S & x(d_1) - y_{VQ1} \end{aligned} \tag{30}$$

gefunden. Der Vektor I_i in der Formel (30) mit 1 ≤ i ≤ S ist ein Einheitsdatenfolgevektor der d_i-Dimension, bei dem alle Elemente gleich 1 sind. Die **Fig.** 35 zeigt ein konkretes Beispiel für diesen Fall.

[0318] Diese restlichen Vektoren $R_1, R_2, ..., R_S$ werden unter Verwendung separater Codebücher vektorquantisiert. Obgleich hier für die Vektorquantisierunq eine gerade Vektorquantisierunq verwendet wird, ist es auch möglich, eine anders strukturierte Vektorquantisierunq zu verwenden. Das heißt, für die folgende Formel (31), in welcher die restlichen Vektoren $R_1, R_2, ..., R_S$ durch Elemente

$$\begin{aligned} R_1 &= (r_1, r_2, ..., r_{d1})^t \\ &\vdots \\ R_i &= (r_{gi+1}, ..., r_{gi+di})^t \end{aligned}$$

(**31**) ausgedrückt sind, sind vektorquantisierte Daten durch $R_{VQ1}, R_{VQ2}, ..., R_{VQS}$ und generell durch R_{VQi} dargestellt, mit

$$R_{VQi} = (r_{VQ}(gi + 1), ..., r_{VQ}(gi + di))^t. \tag{32}$$

[0319] Diese Daten können als die restlichen Vektoren Ri betrachtet werden, an die ein Quantisierungsfehler ε_i angehängt ist. Das heißt, es gilt

$$r_{VQi} = R_i + \varepsilon_i \tag{33}$$

[0320] Das heißt, es gilt

$$\begin{matrix} r_{VQ(gi+1)} & & r_{gi+1} + \varepsilon_{gi+1} \\ & = & \\ R_{VQ(gi+di)} & & r_{gi+di} + \varepsilon_{gi+di} \end{matrix} \tag{34}$$

[0321] Die **Fig.** 36 zeigt ein konkretes Beispiel der Elemente der restlichen Vektoren R_{VQ1}, R_{VQ2}, ..., R_{VQS} nach der Quantisierung.

[0322] Ein auf die Codierungsseite übertragenes Indexausgangssignal ist ein Y_{VQ} anzeigender Index, und S Einheiten aus Indizes zeigen die S Einheiten der restlichen Vektoren R_{VQ1}, R_{VQ2}, ..., R_{VQS} an. Indessen ist bei der Form-Verstärkungs-Vektorquantisierung ein Ausgangsindex durch einen Index zur Formung und einen Index zur Verstärkung dargestellt. Zur Erzeugung eines decodierten Wertes der Vektorquantisierung wird die folgende Operation ausgeführt. Nachdem Y_{VQ}, R_{VQi} mit $1 \le i \le S$ durch ein Tabellennachschlagen aus dem übertragenen Index erhalten sind, wird die folgende Operation ausgeführt. Das heißt, es wird aus der Formel (29) y_{VQi} gefunden, und X_{VQi} wird wie folgt gefunden:

$$\begin{aligned} X_{VQi} &= R_{VQ1} + y_{VQi}\, I_i \quad (1 \le i \le S) \\ &= R_i + \varepsilon_i + y_{VQi}\, I_i \\ &= X_i - y_{VQi}\, I_i + \varepsilon_i + y_{VQi}\, I_i \\ &= X_1 - \varepsilon_i . \end{aligned} \tag{35}$$

[0323] Deshalb ist das in einem Dekodiererausgangssignal erscheinende Quantisierungsrauschen nur das während der Quantisierung von R_i erzeugte ε_i . Die Qualität der Quantisierung von Y auf der ersten Stufe ist nicht direkt im endgültigen Rauschen enthalten. Jedoch beeinflusst eine solche Qualität die Eigenschaften der Vektorquantisierung von RVQi auf der zweiten Stufe, wobei es schließlich auf den Pegel des Quantisierungsrauschens im Decodiererausgangssignal beiträgt.

[0324] Durch die hierarchische Struktur des Codebuchs der Vektorquantisierung wird es möglich

i) die Wiederholungszahl der Multiplikation und Addition zur Codebuchsuche zu reduzieren,

ii) die ROM-Kapazität für das Codebuch zu reduzieren, und

iii) einen effektiven Gebrauch der hierarchischen Fehlerkorrekturcodes zu machen.

[0325] Ein konkretes Beispiel betreffend die Effekte von i) und ii) wird nachfolgend gegeben.

[0326] Es sei nun angenommen, dass M = 44, S = 7, $d_1 = d_2 = d_3 = d_4 = 5$ und $d_5 = d_6 = d_7 = 8$ gilt. Es sei auch angenommen, dass die Zahl der zur Quantisierung der Daten x(n) (= $c_{dB}(n)$) und $1 \le n \le M$ verwendeten Bits gleich 48 ist.

[0327] Wenn der M = 44-dimensionale Vektor mit einem 48-Bit-Ausgangssignal vektorquantisiert wird, ist die Tabellengröße des Codebuchs gleich $2^{48} \approx 2{,}81 \times 10^{14}$. Dies wird dann mit einer Wortbreite (= 44) multipliziert, um annähernd $1{,}238 \times 10^{16}$ zu ergeben, was die Zahl der benötigten Wörter der Tabelle ist. Das Operationsvolumen für Tabellensuche ist ebenfalls ein Wert in der Größenordnung von $2^{48} \times 44$.

[0328] Es sei die folgende Bitzuordnung betrachtet:

Y → 13 Bit (8 Bit: Form, 5 Bit: Verstärkung),

Dimension S = 7

X_1 → 6 Bit, Dimension $d_1 = 5$

X_2 → 5 Bit, Dimension $d_2 = 5$

X_3 → 5 Bit, Dimension $d_3 = 5$

X_4 → 5 Bit, Dimension $d_4 = 5$

X_5 → 5 Bit, Dimension $d_5 = 8$

X_6 → 5 Bit, Dimension $d_6 = 8$

X_7 → 4 Bit, Dimension $d_7 = 8$

Insgesamt: 48 Bit, (M=) 44 Dimensionen.

[0329] Für die Tabellenkapazität zu diesem Zeitpunkt,

Y: Form: $7 \times 2^8 = 1792$, Verstärkung: $2^5 = 32$

X_1: $5 \times 26 = 320$

X_2: 5 × 25 = 160
X_3: 5 × 25 = 160
X_4: 5 × 25 = 160
X_5: 8 × 25 = 256
X_6: 8 × 25 = 256
X_7: 8 × 24 = 128

[0330] Das heißt es sind insgesamt 3264 Wörter erforderlich. Da das Operationsvolumen zur Tabellensuche grundsätzlich von der gleichen Größenordnung wie die Tabellengröße insgesamt ist, ist es in der Ordnung von annähernd 3264. Dieser Wert ist praktisch einwandfrei.

[0331] Was iii) betrifft, so kann ein Verfahren, bei dem die oberen 3, 3, 2, 2, 2 und 1 Bits der Indizes von X_1 bis X_7 geschützt sind und die unteren Bits ohne Fehlerkorrektur verwendet werden, für X_1 bis X_7 zum Schützen der 13 Bits der Quantisierungsausgangsindizes des erststufigen Vektors Y durch eine Vorwärtsfehlerkorrektur (FEC), beispielsweise die Faltungscodierunq, angewendet werden. Eine effektivere FEC kann durch Aufrechterhaltung einer Relation zwischen den binären Daten des den Index des Vektorquantisierers anzeigenden Hammingabstandes und des Euklidabstandes des durch den Index bezeichneten Codevektors, das heißt durch Zuordnen des kleineren Hammingabstandes zum kleineren Euklidabstand des Codevektors angewendet werden.

[0332] Wie klar aus der vorhergehenden Beschreibung zu entnehmen ist, wird gemäß dem oben erwähnten hocheffizienten Codierungsverfahren das strukturierte Codebuch verwendet, und die M-dimensionalen Vektordaten werden in mehrere Gruppen geteilt, um den für jede Gruppe repräsentativen Werte zu finden, wodurch die M-Dimension auf die S-Dimension erniedrigt wird. Dann werden S-dimensionalen Vektordaten mit der ersten Vektorquantisierung verarbeitet, so dass der S-dimensionale Codevektor das lokale Decodiererausgangssignal bei der ersten Vektorquantisierung ist. Der S-dimensionale Codevektor wird auf den ursprünglichen M-dimensionalen Vektor expandiert, wodurch die Daten gefunden werden, welche die Relation zwischen den Daten auf der Frequenzachse des ursprünglichen M-dimensionalen Vektors anzeigen, und dann wird die zweite Vektorquantisierung ausgeführt. Deshalb ist es möglich, das Operationsvolumen für die Codebuchsuche und die Speicherkapazität für das Codebuch zu reduzieren und die Fehlerkorrekturcodierung bei der oberen und unteren Seite der hierarchischen Struktur effektiv anzuwenden.

[0333] Außerdem werden gemäß dem anderen hocheffizienten Codierungsverfahren die Daten auf der Frequenzachse im Voraus nichtlinear komprimiert und dann vektorquantisiert. Auf diese Weise ist es möglich, eine effiziente Codierung zu realisieren und Qualität der Quantisierung zu verbessern.

[0334] Des weiteren wird gemäß dem anderen hocheffizienten Codierungsverfahren die Interblockdifferenz vorhergehender und nachfolgender Blöcke für die auf der Frequenzachse für jeden Block erhaltenen Daten genommen, und die Interblockdifferenzdaten werden vektorquantisiert. Auf diese Weise ist es möglich, das Quantisierungsrauschen weiter zu reduzieren und das Kompressionsverhältnis zu verbessern.

[0335] Indessen wird es hinsichtlich des Stimmhaft/Stimmlos-Grades oder der Tonhöhe der Stimme bzw. Sprache, der bzw, die im Fall der Sprach-Synthese-Analyse-Codierung wie beispielsweise der oben erwähnten MBE schon als charakteristische Volumina bzw. Lautstärkepegel extrahiert sind, möglich, das Codebuch für Vektorquantisierung in Abhängigkeit von diesen charakteristischen Volumina bzw. Lautstärkepegeln, insbesondere den Ergebnissen der Stimmlos/Stimmhaft-Entscheidung umzuschalten. Das heißt, die Spektrumform differiert zwischen dem stimmhaften Ton und dem stimmlosen Ton signifikant, so dass es sehr wünschenswert ist, separat trainierte Codebücher für die jeweiligen Zustände zu haben. Im Fall der hierarchisch strukturierten Vektorquantisierung kann die Vektorquantisierung für die Schicht höherer Ordnung mit einem festen Codebuch ausgeführt werden, während das Codebuch für die Vektorquantisierung der Schicht niedrigerer Ordnung zwischen dem stimmhaften und dem stimmlosen Ton umgeschaltet werden kann. Andererseits kann die Bitzuordnung auf der Frequenzachse umgeschaltet werden, so dass der Ton niedriger Tonhöhe für den stimmhaften Ton betont bzw. hervorgehoben wird, und dass der Ton hoher Tonhöhe für den stimmlosen Ton betont bzw. hervorgehoben wird. Für die Umschaltsteuerung können das Vorhandensein oder die Abwesenheit der Tonhöhe, das Verhältnis bzw. die Proportion des stimmhaften Tons/stimmlosen Tons, der Pegel oder die Neigung des Spektrums usw. verwendet werden.

[0336] Indessen wird in dem Fall der Vektorquantisierung zur Quantisierung mehrerer Daten, die in einem durch einen einzelnen Code anstelle einer separaten Quantisierung von Zeitachsendaten, Frequenzachsendaten und Filterkoeffizientendaten bei der Codierung ausgedrückten Vektor gruppiert sind, das feste Codebuch zur Vektorquantisierung der Spektrumenveloppe der MBE, SBE und LPC oder von Parametern derselben wie beispielsweise ein LSP-Parameter, α-Parameter und k-Parameter verwendet. Jedoch bei Reduzierung der Zahl der verwendbaren Bits, das heißt bei Erniedrigung der Bitrate, wird es unmöglich mit dem festen Codebuch eine ausreichende Leistung zu erhalten. Deshalb ist es wünschenswert, die Eingangsdaten, die durch Gruppierung klassifiziert sind, so dass der Bereich ihrer Präsenz im Vektorraum eingeengt ist, vektorzuquantisieren.

[0337] Es wird in Betracht gezogen, dass selbst bei ausreichend hoher Übertragungsbitrate das strukturierte

Codebuch zur Reduzierung des Operationsvolumens für die Suche verwendet wird. In diesem Fall ist es wünschenswert, anstelle der Verwendung eines einzigen Codebuchs aus (n + 1) Bits das Codebuch in zwei Codebücher zu teilen, deren jedes eine Ausgangsindexlänge von n Bits aufweist.

[0338] Im Hinblick auf den oben erwähnten Stand der Technik ist ein hocheffizientes Codierungsverfahren vorgeschlagen, wodurch es möglich ist, eine effiziente Vektorquantisierung entsprechend den Eigenschaften von Eingangsdaten auszuführen, die Größe des Codebuchs des Vektorquantisierers und das Operationsvolumen für die Suche zu reduzieren und eine Codierung hoher Qualität auszuführen.

[0339] Das hocheffiziente Codierungsverfahren weist die Schritte auf: Finden von Daten auf der Frequenzachse als einen Mdimensionalen Vektor auf der Basis von durch Teilen von Eingangsaudiosignalen wie beispielsweise Sprachsignalen und akustischen Signalen auf der blockweisen Basis und Umwandeln der Signale in Daten auf der Frequenzachse erhaltenen Daten, und Ausführen einer Quantisierung durch Verwendung eines Vektorquantisierers, der abhängig von Zuständen von Audiosignalen mehrere Codebücher zur Ausführung einer Vektorquantisierung bei den Daten auf der Frequenzachse der M-Dimension aufweist, und durch Umschalten und Quantisieren der mehreren Codebücher entsprechend Parametern, die Charakteristiken der Eingangsaudiosignale für jeden Block anzeigen.

[0340] Das andere hocheffiziente Codierungsverfahren weist die Schritte auf: Finden von Daten auf der Frequenzachse als den M-dimensionalen Vektor auf der Basis von durch Teilen von Eingangsaudiosignalen auf der blockweisen Basis und durch Umwandeln der Signale in Daten auf der Frequenzachse erhaltenen Daten, Reduzieren der M-Dimension auf eine S-Dimension, wobei S < M ist, durch Teilen der Daten auf der Frequenzachse der M-Dimension in mehrere Gruppen und durch Finden repräsentativer Werte für jede der Gruppen, Ausführen einer ersten Vektorquantisierung bei den Daten des S-dimensionalen Vektors, Finden eines korrespondierenden S-dimensionalen Codevektors durch inverse Vektorquantisierung der Ausgangsdaten der ersten Vektorquantisierung, Expandieren des S-dimensionalen Codevektors auf den ursprünglichen M-dimensionalen Vektor, und Ausführen einer Quantisierung durch Verwendung eines Vektorquantisierers für die zweite Vektorquantisierung, der abhängig von Zuständen der Audiosignale mehrere Codebücher zur Ausführung einer zweiten Vektorquantisierung bei Daten aufweist, die Relationen zwischen dem expandierten M-dimensionalen Vektor und den Daten auf der Frequenzachse des ursprünglichen M-dimensionalen Vektors anzeigen, und durch Umschalten der mehreren Codebücher entsprechend Parametern, welche Charakteristiken der Eingangsaudiosignale für jeden Block anzeigen.

[0341] Bei der Vektorquantisierung gemäß diesen hocheffizienten Codierungsverfahren ist es bei Verwendung eines Sprachsignals als das Audiosignal möglich, abhängig von einem Stimmhaft/Stimmlos-Zustand des Sprachsignals mehrere Codebücher als das Codebuch zu verwenden, um Parameter, die anzeigen, ob das Eingangssprachsignal für jeden Block stimmhaft oder stimmlos ist, als den Charakteristikparameter zu verwenden. Auch ist es möglich, als Charakteristikparameter den Tonhöhenwert, die Stärke der Tonhöhenkomponente, die Proportion des stimmhaften und stimmlosen Tons, die Neigung und den Pegel des Signalspektrums usw. zu verwenden, und es wird grundsätzlich vorgezogen, das Codebuch abhängig davon umzuschalten, ob das Sprachsignal stimmhaft oder stimmlos ist. Solche Charakteristikparameter können separat übertragen werden, während ursprünglich übertragene Parameter, wie sie durch das Codierungssystem im voraus vorgeschrieben sind, anstelle dessen verwendet werden können. Als die Daten auf der Frequenzachse des M-dimensionalen Vektors können Daten verwendet werden, die auf der blockweisen Basis in Daten auf der Frequenzachse umgewandelt und nichtlinear komprimiert sind. Außerdem kann vor der Vektorquantisierung eine Interblockdifferenz von vektorzuquantisierenden Daten genommen werden, so dass die Vektorquantisierung bei den Interblockdifferenzdaten ausgeführt werden kann.

[0342] Da die Quantisierung durch Umschalten der mehreren Codebücher entsprechend den die Charakteristiken des Eingangsaudiosignals für jeden Block anzeigenden Parametern ausgeführt wird, ist es möglich, eine effektive Quantisierung auszuführen, die Größe des Codebuchs des Vektorquantisierers und das Operationsvolumen für jede Suche zu reduzieren und eine Codierung hoher Qualität auszuführen.

[0343] Eine Ausführungsform des hocheffizienten Codierungsverfahrens wird nachfolgend unter Bezugnahme auf die Zeichnungen erläutert.

[0344] Die **Fig.** 39 zeigt eine schematische Anordnung eines Codierers zur Darstellung des hocheffizienten Codierungsverfahren als eine Ausführungsform der vorliegenden Erfindung.

[0345] Bei der **Fig.** 39 wird ein Eingangssignal, beispielsweise ein Sprachsignal oder ein akustisches Signal, einem Eingangsanschluss **711** zugeführt und dann von einem Frequenzachsenumwandlungsabschnitt **712** in Spektrumamplitudendaten auf der Frequenzachse umgewandelt. Im Frequenzachsenumwandlungsabschnitt **712** sind ein Blockbildungsabschnitt **712a** zum Teilen des Eingangssignals auf der Zeitachse in Blöcke, deren jeder eine vorbestimmte Zahl Abtastwerte, beispielsweise N Abtastwerte aufweist, ein Orthogonaltransformationsabschnitt **712b** für schnelle Fouriertransformation (FFT) usw., und ein Datenprozessor **712c** zum Finden von Charakteristiken der Spektrumenveloppe anzeigenden Amplitudendaten vorgesehen. Ein Ausgangssignal aus dem Frequenzachsenumwandlungsabschnitt **712** wird über einen fakultativen Nichtlinearkompressor **713** zur Umwandlung in beispielsweise einen dB-Bereich und über einen fakultativen Prozessor zum Nehmen der

Interblockdifferenz zu einem Vektorquantisierungsabschnitt **715** übertragen. Durch den Vektorquantisierungsabschnitt **715** werden eine vorbestimmte Zahl, beispielsweise M Abtastwerte der Eingangsdaten als der M-dimensionale Vektor gruppiert und mit einer Vektorquantisierung verarbeitet. Generell wird bei der M-dimensionalen Vektorquantisierungsverarbeitung das Codebuch für einen Codevektor mit dem kürzesten Abstand vom dimensionalen Eingangsvektor im M-dimensionalen Raum abgesucht, und der Index des Codevektors, nach dem gesucht wird, wird von einem Ausgangsanschluss **716** ausgegeben. Der Vektorquantisierungsabschnitt **715** der in **Fig.** 39 gezeigten Ausführungsform enthält mehrere Arten Codebücher, die entsprechend Charakteristiken des Eingangssignals aus dem Frequenzachsenumwandlungsabschnitt **712** umgeschaltet werden.

[0346] Beim Beispiel der **Fig.** 39 ist angenommen, dass das Eingangssignal ein Sprachsignal ist. Ein Stimmhaftcodebuch (V-Codebuch) $\mathbf{715_V}$ und ein Stimmloscodebuch $\mathbf{715_U}$ werden von einem Umschalter $\mathbf{715_W}$ umgeschaltet und zu einem Vektorquantisierer $\mathbf{715_Q}$ übertragen. Der Umschalter $\mathbf{715_W}$ wird entsprechend einem Stimmhaft/Stimmlos-Entscheidungssignal (V/UV-Entscheidungssignal) aus dem Frequenzaehsenumwandlungsabschnitt **712** gesteuert. Das V/W-Signal oder -Kennzeichen ist ein Parameter, der im Fall eines später beschriebenen Multibanderregungsvocoders (MBE-Vocoder) (Sprach-Analyse-Synthese-Einrichtung) von der Analyseseite (Codierer) zur Syntheseseite (Decodierer) zu übertragen ist und nicht separat übertragen zu werden braucht.

[0347] Bezüglich des Beispiels des MBE kann das V/UV-Entscheidungskennzeichen als eine Art der übertragenen Daten für den Parameter zum Umschalten der Codebücher $\mathbf{715_V}$, $\mathbf{715_U}$ verwendet werden. Das heißt, der Frequenzachsenumwandlungsabschnitt **712** führt eine Bandteilung entsprechend der Tonhöhe aus und trifft eine V/UV-Entscheidung für jedes der geteilten Bänder. Es sei angenommen, dass die Zahl V-Bänder und die Zahl UV-Bänder gleich N_V bzw. N_{UV} ist. Wenn N_V und N_{UV} mit einer vorbestimmten Schwelle V_{th} die folgende Relation

$$\frac{N_V}{N_V + N_{UV}} \geq V_{th} \qquad (36)$$

erfüllen, wird das V-Codebuch $\mathbf{7I5_V}$ gewählt. Andernfalls wird das UV-Codebuch $\mathbf{175_U}$ gewählt. Die Schwell V_{th} kann auf beispielsweise etwa 1 gesetzt werden.

[0348] Auf der Decodiererseite (Syntheseseite) wird ebenfalls das Umschalten und die Wahl der zwei Arten von V- und UV-Codebüchern ausgeführt. Beim MBE-Vocoder ist es, da das V/UV-Entscheidungskennzeichen eine in jedem Fall zu übertragende Neben- bzw. Seiteninformation ist, nicht erforderlich, separate charakteristische Parameter für das Codebuchumschalten bei diesem Beispiel zu übertragen, wodurch keine Erhöhung der Übertragungsbitrate verursacht wird.

[0349] Die Erzeugung oder das Training des V-Codebuchs $\mathbf{715_V}$ und des UV-Codebuchs $\mathbf{716_U}$ wird einfach durch Teilen von Trainingsdaten durch die gleichen Standards möglich gemacht. Das heißt, es wird angenommen, dass ein von der Gruppe von Amplitudendaten, bei denen festgestellt ist, dass sie stimmhaft (V) sind, erzeugtes Codebuch das V-Codebuch $\mathbf{715_V}$ ist, und dass ein von der Gruppe von Amplitudendaten, bei denen festgestellt ist, dass sie stimmlos (W) sind, erzeugtes Codebuch das UV-Codebuch $\mathbf{715_U}$ ist.

[0350] Bei dem vorliegenden Beispiel ist es, da die V/UV Information zum Umschalten des Codebuchs verwendet wird, notwendig, das V/UV-Kennzeichen zu sichern, das heißt, eine hohe Zuverlässigkeit des V/UV-Kennzeichens zu haben. Beispielsweise sollten in einem klar als ein Konsonant oder ein Hintergrundrauschen betrachteten Abschnitt alle Bänder W sein. Als ein Beispiel der obigen Entscheidung sei darauf hingewiesen, dass winzige oder kleine Eingangssignale hoher Leistung im Hochfrequenzbereich gleich W gemacht werden.

[0351] Die schnelle Fouriertransformation (FFT) wird bei den N Punkten des Eingangssignals (**256** Abtastwerte) ausgeführt, und in jedem der Abschnitte von 0 bis N/4 und N/4 bis N/2 zwischen effektiv 0 bis π (0 bis N/2) wird eine Leistungsberechnung ausgeführt.

$$P_L = \sum_{i=0}^{(N/4)-1} rms^2(i)$$

$$P_H = \sum_{i=N/4}^{(N/2)-1} rms^2(i), \qquad (37)$$

wobei rms(i) gleich

$$\sqrt{\mathrm{Re}^2(i) + \mathrm{Im}^2(i)}$$

ist und wobei Re (i) und Im (i) der Realteil bzw. Imaginärteil der FFT der Eingangsprogression ist. Bei Verwendung von P_L und PH der Formel (37) wird die folgende Formel

$$R_d = \frac{P_L}{P_H}$$

$$L = \sqrt{\frac{P_L + P_H}{N/2}}$$

(38)

erzeugt. Wenn Rd < R_{th} und L < L_{th} gilt, werden alle Bänder bedingungslos W gemacht.

[0352] Diese Operation hat den Effekt der Vermeidung der Verwendung einer im winzigen oder kleinen Eingangssignal detektierten falschen Tonhöhe. Auf diese Weise ist die Erzeugung eines sicheren V/UV-Kennzeichens von vorne herein zum Umschalten des Codebuchs bei der Vektorquantisierung sollten.

[0353] Als nächstes wird das Training bei der Erzeugung der Vund U/V-Codebücher unter Bezugnahme auf die **Fig.** 40 erläutert.

[0354] Bei der **Fig.** 40 wird ein Signal aus einem Trainingssatz **731**, der aus einem Trainingssprachsignal für mehrere Minuten besteht, zu einem Frequenzachsenumwandlungsabschnitt **732** gesendet, wo eine Tonhöhenextraktion von einem Tonhöhenextraktionsabschnitt **732a** ausgeführt wird und eine Berechnung der Spektrumamplitude von einem Spektrumamplitudenberechnungsabschnitt **732b** ausgeführt wird. Auch wird für jedes Band eine V/UV-Entscheidung von einem V/UV-Entscheidungsabschnitt **732c** für jedes Band getroffen. Ausgangsdaten aus dem Frequenzachsenumwandlungsabschnitt **732** werden zu einem Vortrainingsverarbeitungsabschnitt **734** übertragen.

[0355] Im Vortrainingsverarbeitungsabschnitt **734** werden die Bedingungen der Formeln (36) und (38) von einem Prüfabschnitt **334a** geprüft, und entsprechend der resultierenden V/UV-Information werden die Spektrumamplitudendaten von einem Trainingsdatenzuordnungsabschnitt **734b** zugeordnet. Die Amplitudendaten werden zu einem V- Trainingsausgangsdatenabschnitt **736a** für stimmhafte Töne (V-Töne) und zu einem UV-Trainingsdatenausgangsabschnitt **737a** für stimmlose Töne (UV-Töne) übertragen.

[0356] Die vom V-Trainingsdatenausgabeabschnitt **736a** ausgegebenen V-Spektrumamplitudendaten werden zu einem Trainingsprozessor **736b** gesendet, bei dem eine Trainingsverarbeitung durch beispielsweise das LBG-Verfahren ausgeführt wird, wodurch ein V-Codebuch **736c** erzeugt wird. Das LBG-Verfahren ist ein Trainingsverfahren für das Codebuch in einem Algorithmus zum Konstruieren eines Vektorquantiserers, das in Linde, Y., Buzo, A. und Gray, R. M., „An Algorithm for Vector Quantizer Design", IEEE Trans. Comm., COM-28, Jan. 1980, Seiten 84–95 vorgeschlagen ist. Dieses LBG-Verfahren dient zum Konstruieren eines lokal optimalen Vektorquantisierers durch Verwendung einer sogenannten Trainingskette für eine Informationsquelle mit einer unbekannten Wahrscheinlichkeitsdichtefunktion. Ähnlich werden die vom W-Trainingsdatenausgabeabschnitt **737a** ausgegebenen UV-Spektrumamplitudendaten zu einem Trainingsprozessor **737c** gesendet, bei dem eine Trainingsverarbeitung durch beispielsweise das LBG-Verfahren ausgeführt wird, wodurch ein UV-Codebuch **737c** erzeugt wird.

[0357] Weist der Vektorquantierungsabschnitt eine hierarchische Struktur auf, bei der ein Codebuch eines Abschnitts für gemeinsamen V/UV-Gebrauch für die obere Schicht verwendet wird, während nur das Codebuch für die untere Schicht entsprechend V/UV umgeschaltet wird, was später zu beschreiben ist, ist es notwendig, das Codebuch eines Abschnitts für gemeinsamen V/UV-Gebrauch zu erzeugen. In diesem Fall ist es notwendig, die Ausgangsdaten aus dem Frequenzachsenumwandlungsabschnitt **732** zu einem Trainingsdatenausgabeabschnitt **735a** für ein Codebuch eines Abschnitts für gemeinsamen V/UV-Gebrauch zu senden. Die vom Trainingsdatenausgabeabschnitt **735a** für das Codebuch des Abschnitts für gemeinsamen V/UV-Gebrauch ausgegebenen Spektrumamplitudendaten werden zu einem Trainingsprozessor **735b** gesendet, wo eine Trainingsverarbeitung durch beispielsweise das LBG-Verfahren ausgeführt wird, wodurch ein Codebuch **735c** für gemeinsamen V/UV-Gebrauch erzeugt wird. Es ist erforderlich, den Codevektor vom erzeugten Codebuch **735c** für gemeinsamen V/UV-Gebrauch zum V- Trainingsdatenausgabeabschnitt **736a** und zum UV-Trainingsdatenausgabeabschnitt **737a** zu senden, eine Vektorquantisierung für die obere Schicht an den V- und UV-Trainingsdaten durch Verwendung des Codebuchs für gemeinsamen V/UV-Gebrauch auszuführen und V- und UV-Trainingsdaten für die untere Schicht zu erzeugen.

[0358] Eine konkrete Anordnung und Operation der hierarchisch strukturierten Vektorquantisierungseinheit wird unter Bezugnahme auf die **Fig.** 41 und die **Fig.** 31 bis 36 erläutert. Die in **Fig.** 41 gezeigte Vektorquantisierungseinheit **715** ist so hierarchisch strukturiert, dass sie zwei Schichten aufweist, beispielsweise eine obere und eine untere Schicht, bei denen eine zweistufige Vektorquantisierung am Eingangsvektor ausgeführt wird, wie es bezüglich der **Fig.** 31 bis 36 erläutert ist.

[0359] Die Rmplitudendaten auf der Frequenzachse aus dem Frequenzachsenumwandlungsabschnitt **712** nach **Fig.** 39 werden über den fakultativen Nichtlinearkompressor **713** und den fakultativen Interblockdifferenz-Verarbeitungsabschnitt 714 einem Eingangsanschluss **717** der in **Fig.** 41 gezeigten Vektorquantisierungseinheit **715** als der M-dimensionale Vektor zugeführt, der die Einheit für die Vektorquantisierung zu sein hat. Der Mdimensionale Vektor wird zu einem Dimensionsreduktionsabschnitt **721** übertragen, wo er in meh-

rere Gruppen geteilt wird und seine Dimension durch wie in den **Fig.** 31 und 32 gezeigtes Finden des repräsentativen Wertes für jede der Gruppen auf eine S-Dimension (S < M) reduziert wird.

[0360] Als nächstes wird der S-dimensionale Vektor durch einen S-dimensionalen Vektorquantisierer **722$_Q$** quantisiert. Das heißt, unter den S-dimensionalen Codevektoren in einem Codebuch **722$_C$** des S-dimensionalen Vektorquantisierers **722$_Q$** wird das Codebuch nach dem Codebuch mit dem kürzesten Abstand vom S-dimensionalen Eingangsvektor im S-dimensionalen Raum abgesucht, und die Indexdaten des gesuchten Codevektors werden von einem Ausgangsanschluss **726** ausgegeben. Der gesuchte Codevektor (ein durch inverse Vektorquantisierung des Ausgangsindex erhaltener Codevektor) wird zu einem Dimensionsexpandierungsabschnitt **723** gesendet. Für das Codebuch **722$_C$** wird das in **Fig.** 40 erläuterte Codebuch 735_C für gemeinsamen V/UV-Gebrauch wie in **Fig.** 33 gezeigt verwendet. Der Dimensionsexpandierungsabschnitt **723** expandiert wie in der **Fig.** 34 gezeigt den S-dimensionalen Codevektor auf den ursprünglichen M-dimensionalen Vektor.

[0361] Bei dem Beispiel nach **Fig.** 41 werden die expandierten M-dimensionalen Vektordaten aus dem Dimensionsexpandierungsabschnitt 723 einem Subtrahierer **724** zugeführt, wo S Einheiten von Vektoren, die Relationen zwischen dem vom S-dimensionalen Vektor expandierten M-dimensionalen Vektor und dem ursprünglichen M-dimensionalen Vektor anzeigen, wie in **Fig.** 35 gezeigt durch Subtrahieren von den Daten auf der Frequenzachse des ursprünglichen M-dimensionalen Vektors erzeugt werden.

[0362] Die auf diese Weise vom Subtrahierer **724** erhaltenen S Vektoren werden jeweils mit einer Vektorquantisierung durch je eine von S Einheiten von Vektorquantisierern **725$_{1Q}$** bis **725$_{SQ}$** einer Vektorquantisierungsgruppe **725** verarbeitet. Von den Vektorquantisierern **725$_{1Q}$** bis **725$_{SQ}$** ausgegebene Indizes werden von jeweiligen Ausgangsanschlüssen **727$_{1Q}$** bis **727$_{SQ}$** wie in **Fig.** 36 gezeigt ausgegeben.

[0363] Für die Vektorquantisierer **725$_{1Q}$** bis **725$_{SQ}$** werden jeweilige V-Codebücher **725$_{1V}$** bis **725$_{SV}$** und jeweilige UV-Codebücher **725$_{1U}$** bis **725$_{SQ}$** verwendet. Diese V-Codebücher **725$_{1V}$** bis **725$_{SV}$** und UV-Codebücher **725$_{1U}$** bis **725$_{SU}$** werden von entsprechend einer V/UV-Information aus einem Eingangsanschluss **718** gesteuerten Umschaltern **725$_{1W}$** bis **725$_{SW}$** umgeschaltet, um gewählt zu werden. Diese Umschalter **725$_{1W}$** bis **725$_{SW}$** können für alle Bänder gleichzeitig oder sperrend zum Umschalten gesteuert werden. Jedoch hinsichtlich der verschiedenen Frequenzbänder der Vektorquantisierer **725$_{1Q}$** bis **725$_{SQ}$** können die Umschalter **725$_{1W}$** bis **725$_{SW}$** entsprechend einem V/UV-Kennzeichen für jedes Band zur Umschaltung gesteuert werden. Es ergibt sich von selbst, dass die V-Codebücher **725$_{1V}$** bis **725$_{SV}$** mit dem V-Codebuch **736c** in **Fig.** 40 korrespondieren, und dass die UV-Codebücher **725$_{1U}$** bis **725$_{SU}$** mit dem UV-Codebuch **737c** korrespondieren.

[0364] Durch Ausführen der hierarchisch strukturierten zweistufigen Vektorquantisierung wird es möglich, das Operationsvolumen der Codebuchsuche zu reduzieren und das Speichervolumen (beispielsweise ROM-Kapazität) für das Codebuch zu reduzieren. Auch wird es durch Ausführen einer Fehlerkorrekturcodierung an einem vom Ausgangsanschluss **726** erhaltenen wichtigeren Index auf der oberen Schicht möglich, den Fehlerkorrekturcode effektiv anzunehmen. Indessen ist die hierarchische Struktur der Vektorquantisierungseinheit **715** nicht auf die zwei Stufen beschränkt, sondern sie kann eine Multischichtstruktur aus drei oder mehr Stufen sein.

[0365] Jeder Abschnitt der **Fig.** 39 bis 41 muss nicht ganz durch Hardware gebildet sein, sondern kann unter Verwendung beispielsweise eines Digitalsignalprozessors (DSP) durch Software realisiert werden.

[0366] Wie oben beschrieben kann beispielsweise in dem Fall der Sprach-Synthese-Analyse-Codierung hinsichtlich des Stimmhaft/Stimmlos-Grades und der Tonhöhe, die im voraus als die charakteristischen Volumina bzw. Lautstärkepegel extrahiert sind, eine gute Vektorquantisierung durch Umschalten des Codebuches entsprechend den charakteristischen Größen, insbesondere des Ergebnisses der Stimmhaft/Stimmlos-Entscheidung realisiert werden. Das heißt, die Form des Spektrums differiert stark zwischen dem stimmhaften Ton und dem stimmlosen Ton, und infolgedessen wird es im Sinne einer Verbesserung der Charakteristiken sehr vorgezogen, die Codebücher entsprechend den jeweiligen Zuständen separat zu trainieren. Auch kann in dem Fall der hierarchisch strukturierten Vektorquantisierung ein festes Codebuch zur Vektorquantisierung der oberen Schicht verwendet werden, während ein Umschalten von zwei Codebüchern, das heißt eines stimmhaften und stimmlosen Codebuches nur für die Vektorquantisierung auf der unteren Schicht verwendet werden kann. Auch kann bei der Bitzuordnung auf der Frequenzachse das Codebuch gewechselt werden, so dass der niedrigtonige Klang bzw. Ton für den stimmhaften Klang bzw. Ton betont bzw. hevorgehoben wird, während der hochtonige Klang bzw. Ton für den stimmlosen Klang bzw. Ton betont hervorgehoben wird. Für die Umschaltsteuerung können das Vorhandensein oder Fehlen der Tonhöhe, das Stimmhaft/Stimmlos-Verhältnis, der Pegel und die Neigung des Spektrums usw. verwendet werden. Außerdem können drei oder mehr Codebücher umgeschaltet werden. Beispielsweise können zwei oder mehr stimmlose Codebücher für Konsonanten und für Hintergrundrauschen usw. verwendet werden.

[0367] Als nächstes wird ein konkretes Beispiel des Vektorquantisierungsverfahrens, bei dem eine Quantisierung durch Gruppierung der Wellenform des Klanges bzw. Tones und der mehreren Abtastwerte der Spektrumenveloppeparameter in einem durch einen einzelnen Code ausgedrückten Vektor ausgeführt wird, erläutert.

[0368] Die oben erwähnte Vektorquantisierung dient zum Ausführen einer **Abb.** Q von einem in einem k-dimensionalen Euklidraum R^k vorhandenen Eingangsvektor X in einen Ausgangsvektor y. Der Ausgangsvektor y ist aus einer Gruppe von N Einheiten von Wiedergabevektoren $Y = \{y_1, y_2, ..., y_N\}$ gewählt. Das heißt, der

Ausgangsvektor y kann durch

$$Y = Q(X) \tag{39}$$

ausgedrückt werden, wobei $y \in Y$ gilt. Der Satz Y wird als Codebuch bezeichnet, das N Einheiten (Pegel) aus Codevektoren $Y_1, Y_2, ..., Y_N$ aufweist. Dieses N wird als die Codebuchgröße bezeichnet.

[0369] Beispielsweise hat ein N-pegeliger, k-dimensionaler Vektorquantisierer einen partiellen Raum des aus N Einheiten von Bereichen oder Zellen bestehenden Eingangsraums. Die N Zellen werden durch $\{R_1, R_2,...,R_N\}$ ausgedrückt. Die Zelle R_i ist beispielsweise ein Satz aus einem y_i als den repräsentativen Vektor wählenden Eingangsvektor X und kann durch

$$R_i = Q^{-1}(y_i) = \{x\ R^k: Q(x) = y_i\} \tag{40}$$

ausgedrückt werden, wobei $1 \le i \le N$ gilt.

[0370] Die Summe aller geteilten Zellen korrespondiert mit dem ursprünglichen k-dimensionalen Euklidraum R^k, und diese Zellen haben keinen überlappten Abschnitt. Dies wird durch die folgende Formel

$$\bigcup_{i=1}^{N} = R^k, R_i \cap R_j = 0, \quad \text{für i} \neq \text{j} \tag{41}$$

ausgedrückt. Demgemäss bestimmt die mit dem Ausgangssatz Y korrespondierende Zellenteilung $\{R_i\}$ den Vektorquantisierer Q.

[0371] Es ist möglich, in Betracht zu ziehen, dass der Vektorquantisierer in einen Codierer C und einen Decodierer De geteilt ist. Der Codierer C führt die Abbildung des Eingangsvektors X auf einen Index i aus. Des Index i wird aus einem Satz von N Einheiten $I = \{1, 2, ... N\}$ gewählt und durch

$$I = C(X) \tag{42}$$

ausgedrückt, wobei $i \in I$ gilt.

[0372] Der Decodierer De führt die Abbildung des Index i auf einen korrespondierenden Wiedergabevektor (Ausgabevektor) y_i aus. Der Wiedergabevektor y_i wird aus dem Codebuch Y gewählt. Dies wird durch

$$Y_i = De(i) \tag{43}$$

ausgedrückt, wobei y_i ε Y gilt.

[0373] Die Operation des Vektorquantisierers ist die der Kombination des Codierers C und des Decodierers De und kann durch die Formeln (39), (49), (41), (42) und (43) und die folgende Formel (44)

$$y = Q(X) = De(i) = De(C(X)) \tag{44}$$

ausgedrückt werden.

[0374] Der Index i ist eine Binärzahl, und die Bitrate Bt als die Transmissionsrate des Vektorquantisierers und die Auflösung b des Vektorquantisierers sind durch die folgenden Formeln

$$Bt = \log_2 N \text{ (bit/Vektor)} \tag{45}$$

$$B = Bt/k \text{ (bit/Abtastwert)} \tag{46}$$

ausgedrückt.

[0375] Als nächstes wird ein Verzerrungsmaß als die Auswertungsskala eines Fehlers erläutert.

[0376] Das Verzerrungsmaß d(X, y) ist eine den Grad der Diskrepanz (Fehler) zwischen dem Eingangsvektor X und dem Ausgangsvektor y anzeigende Skala. Das Verzerrungsmaß d(X, y) ist durch

$$d(X,y) = |X,y|^2 = (X-y)'(X-y) = \sum_{i=1}^{k}(X_i - y_i)^2 \tag{47}$$

ausgedrückt, wobei X_i, y_i die i-ten Elemente des Vektors X bzw. y sind.

[0377] Das heißt, die Leistung des Vektorquantisierers ist definiert durch die gesamte mittlere Verzerrung, die durch

$$Da = E[d(X, y)] \tag{48}$$

gegeben ist, wobei E der Erwartungswert ist.

[0378] Normalerweise zeigt die Formel (48) den Mittelwert einer Zahl Abtastwerte an und kann durch

$$Da = \lim_{M\to\infty} 1/M \sum_{n=1}^{M} d(X_n, y_n)$$

(49)ausgedrückt werden, wobei $\{X_n\}$ ein Eingangsvektorarray mit $y_n = Q(X_n)$ ist. M ist die Zahl Abtastwerte.

[0379] Als nächstes wird der zur Erzeugung des Codebuchs des Vektorquantisierers verwendete LBG-Algorithmus erläutert.

[0380] Ursprünglich ist es schwierig ein konkretes Design des Codebuchs des Vektorquantisierers ohne Kenntnis des Verzerrungsmaßes und der Wahrscheinlichkeitsdichtefunktion (PDF) der Eingangsdaten auszuführen. Jedoch macht es die Verwendung von Trainingsdaten möglich, das Codebuch des Vektorquantisierers ohne die PDF zu bilden. Beispielsweise ist es mit der Dimension k der Codebuchgröße N und den bestimmten Trainingsdaten x(n) möglich, das optimale Codebuch aus diesen Elementen zu erzeugen. Dieses Verfahren ist ein als das LBG-Verfahren bezeichneter Algorithmus. Das heißt, auf der Annahme, dass Trainingsdaten aller Größenarten die PDF der Stimme bzw. Sprache ausdrücken, ist es möglich, ein Codebuch des Vektorquantisierers durch Optimierung der Trainingsdaten zu erzeugen.

[0381] Die Charakteristiken des LBG-Algorithmus bestehen aus der Wiederholung der Nächstnachbarbedingung (optimale Teilungsbedingung) zur Teilung und der Schwerpunktbedingung (Repräsentativpunktbedingung) zur Bestimmung eines repräsentativen Punktes. Das heißt, der LBG-Algorithmus fokussiert darauf, wie die Teilung und der repräsentative Punkt zu bestimmen sind. Die optimale Teilungsbedingung bedeutet die Bedingung für den optimalen Codierer zum Zeitpunkt, bei dem der Decodierer vorgesehen ist. Die Repräsentativpunktbedingung bedeutet die Bedingung für den optimalen Decodierer zum Zeitpunkt, wenn der Codierer vorgesehen ist.

[0382] Unter der optimalen Teilungsbedingung wird die Zelle R_j durch die folgende Formel

$$R_j = \{X : d(X, y_j) \leq d(X, y_i) \text{ für alle } i \neq j,\ i,j \in I\} \quad (50)$$

ausgedrückt, wenn der repräsentative Punkt vorhanden ist. In der Formel (50) ist die j-te Zelle R_j ein Satz Eingangssignal X derart, dass der j-te repräsentative y_i der nächste ist.

[0383] Kurz ausgedrückt bestimmt der Satz Eingangssignal X so, dass der nächste repräsentative Punkt gesucht wird, wenn das Eingangssignal vorhanden ist, den Raum R_j, der den repräsentativen Punkt bildet. In anderen Worten ausgedrückt ist dies eine Operation zum Wählen des Codevektors, der dem gegenwärtigen Eingangssignal in das Codebuch am nächsten ist, das heißt die Operation des Vektorquantisierers oder die Operation des Codierers selbst.

[0384] Wenn der Decodierer wie oben beschrieben bestimmt ist, kann der optimale Codierer so gefunden werden, dass er die minimale Verzerrung ergibt. Der Codierer C wird

$$C(X) = j \text{ iff } d(X, y_j) < d(X, y_i) \text{ für alle } I \varepsilon I, \quad (51)$$

wobei „iff" bedeutet „solange wie ...". Dies bedeutet, dass der Index j ausgegeben wird, wenn der Abstand zwischen den Eingangssignalen X und yj kürzer als der Abstand von jedem y_i ist. Das heißt es ist der optimale Codierer, der den nächsten repräsentativen Punkt findet, und er gibt dessen Index aus.

[0385] Die Repräsentativpunktbedingung ist eine Bedingung, unter welcher bei Bestimmung eines Raumes R_i, das heißt bei Feststellung des Codierers der optimale Vektor y_1 der Schwerpunkt im Raum der i-ten Zelle R_i ist und angenommen ist, dass der Schwerpunkt der repräsentative Vektor ist. Dieser y_1 wird wie folgt bezeichnet:

$$Y_1 = \text{cent}(R_i)\ \varepsilon\ R_i. \quad (52)$$

[0386] Jedoch wird der Schwerpunkt von R_i, das heißt cent(R_i) wie folgt definiert:

$$Y_C = \text{cent}(R_i) \quad (53)$$

wenn

$$\text{wenn } E[d(X, y_C) | X \in R_i] \leq E[d(X, y) | X \in R_i] \text{ für alle } y \in R_i.$$

[0387] Diese Formel (53) zeigt an, dass y_C der repräsentative Punkt im Raum R_i wird, wenn der Erwartungswert der Verzerrung zwischen dem Eingangssignal X im Raum und y_C minimiert wird. Der optimale Codevektor y_i minimiert die Verzerrung im Raum R_i. Demgemäss gibt bei Feststellung des Codierers der optimale Codierer den repräsentativen Punkt des Raumes aus und kann durch die folgende Formel (54)

$$De(i) = cent\ (R_i) \qquad (54)$$

ausgedrückt werden. Normalerweise wird angenommen, dass der Mittelwert (gewichteteter Mittelwert oder einfaches Mittel) des Eingangsvektors X der repräsentative Punkt ist.

[0388] Bei Feststellung der Nächstnachbarbedingung und der Repräsentativpunktbedingung zur Bestimmung der Teilung bzw. des repräsentativen Punktes wird der LBG-Algorithmus entsprechend einem in **Fig.** 43 gezeigten Flussdiagramm ausgeführt.

[0389] Zuerst wird beim Schritt 5821 eine Initialisierung ausgeführt. Speziell wird die Verzerrung D_{-1} auf unendlich eingestellt und dann wird die Iterationszahl n auf „0" (n = 0) gesetzt. Auch werden Y_0, ε und n_m als das anfängliche Codebuch, die Schwelle bzw. die maximale Iterationszahl definiert.

[0390] Beim Schritt 5822 werden mit dem beim Schritt 5821 bereitgestellten anfänglichen bzw, initialen Codebuch Y_0 werden die Trainingsdaten unter der Nächstnachbarbedingung codiert. Kurz ausgedrückt wird das initiale Codebuch durch Abbildung verarbeitet.

[0391] Beim Schritt 5823 wird eine Verzerrungsberechnung zur Berechnung der Quadratsumme des Abstandes zwischen den Eingangsdaten und den Ausgangsdaten ausgeführt.

[0392] Beim Schritt S824 wird festgestellt, ob die beim Schritt 5823 aus der vorhergehenden Verzerrung D_{n-1} und der gegenwärtigen Verzerrung D_n gefundene Reduktionsrate der Verzerrung kleiner als der Schwellenwert ε ist, oder ob die Iterationszahl n die im voraus festgestellte maximale Iterationszahl n_m erreicht hat. Wenn JA gewählt ist, endet die Ausführung des LBG-Algorithmus, und wenn NEIN gewählt ist, geht die Operation zum nächsten Schritt 5825 vor.

[0393] Der Schritt 5825 dient zur Vermeidung, dass der Codevektor mit den Eingangsdaten insgesamt nicht durch Abbildung verarbeitet wird, die im Fall eines unrichtigen initialen Codebuchs beim Schritt 5821 gesetzt wird. Normalerweise wird der Codevektor mit den insgesamt nicht abgebildeten Eingangsdaten in die Nähe einer Zelle bewegt, welche die größte Verzerrung aufweist.

[0394] Beim Schritt 826 wird ein neuer Schwerpunkt durch Berechnung gefunden. Speziell wird der Mittelwert der in der bereitgestellten Zelle vorhandenen Trainingsdaten als ein neuer Codevektor berechnet, der dann aktualisiert wird.

[0395] Die zum Schritt 5827 vorgehende Operation kehrt zum Schritt 5822 zurück, und dieser Operationsfluss wird wiederholt, bis beim Schritt 5824 JA gewählt wird.

[0396] Es ergibt sich, dass der oben erwähnte Fluss den LBG-Algorithmus in einer Richtung zur Verkleinerung der Verzerrung zwischen dem Eingangssignal und dem Ausgangssignal konvergiert, um die Operation bei einer gewissen Stufe aufzuhängen.

[0397] Indessen hat der konventionelle LBG-Algorithmus beim trainierten Vektorquantisierer keine Relation zwischen dem Euklidabstand des Codevektors und dem Hammingabstand seines Index gegeben. Deshalb besteht die Gefahr, dass wegen Codefehlern im Übertragungspfad ein irrelevantes Codebuch gewählt wird.

[0398] Obgleich andererseits ein Einstellverfahren zur Vektorquantisierung im Hinblick auf den Codefehler im Übertragungspfad vorgeschlagen ist, hat es einen Nachteil wie beispielsweise eine Verschlechterung von Charakteristiken bei der Abwesenheit von Fehlern.

[0399] Infolgedessen wird im Hinblick auf den oben beschriebenen Stand der Technik ein Vektorquantisierungsverfahren vorgeschlagen, das Stärke gegen die Übertragungspfadfehler ohne Verursachung einer Verschlechterung von Charakteristiken bei der Abwesenheit der Fehler aufweist.

[0400] Gemäß einem ersten Aspekt der vorliegenden Erfindung ist ein Vektorquantisierungsverfahren zum Suchen eines aus mehreren M-dimensionalen Codevektoren mit M Einheiten von Daten als M Vektoren bestehenden Codebuchs und zur Ausgabe eines Index eines Codebuchs, nach dem gesucht wird, bereitgestellt, wobei das Verfahren koinzidente Größenrelationen eines Abstandes zwischen Codevektoren im Codebuch und einem Hammingabstand mit dem auf binäre Weise ausgedrückten Index aufweist.

[0401] Gemäß dem zweiten Aspekt der vorliegenden Erfindung ist auch ein Vektorquantisierungsverfahren zum Suchen eines aus mehreren M-dimensionalen Codevektoren mit M Einheiten von Daten als M Vektoren bestehenden Codebuch und zum Ausgeben eines Index eines Codebuchs, nach dem gesucht wird, bereitgestellt, wobei ein Teil von Bits von den Index ausdrückenden binären Daten mit einem Fehlerkorrekturcode geschützt ist, und Größenrelationen eines Hammingabstandes zwischen verbleibenden Bits und einem Abstand zwischen Codevektoren im Codebuch miteinander koinzidieren.

[0402] Gemäß dem dritten Aspekt der vorliegenden Erfindung ist außerdem das Vektorquantisierungsverfahren bereitgestellt, bei dem ein durch Gewichtung mit einer zum Definieren eines Verzerrungsmaßes verwendeten gewichteten Matrix gefundener Abstand als ein Abstand zwischen den Codevektoren verwendet wird.

[0403] Mit dem Vektorquantisierungsverfahren des ersten Aspekts der vorliegenden Erfindung ist es dadurch, dass man koinzidente Größenrelationen eines Abstandes zwischen Codevektoren in dem aus mehreren M-dimensionalen Codevektoren mit M Einheiten von Daten als die M-dimensionalen Vektoren bestehenden Codebuch und eines Hammingabstandes mit dem auf binäre Weise ausgedrückten Index des gesuchten Codevektors hat, möglich, Effekte des Codefehlers im Übertragungspfad zu verhindern.
[0404] Mit dem Vektorquantisierungsverfahren des zweiten Aspekts der vorliegenden Erfindung ist es durch Schützen eines Teils von Bits von den Index des gesuchten Codevektors ausdrückenden binären Daten mit einem Fehlercorrekturcode und dadurch, dass man die koinzidenten Größenrelationen eines Hammingabstandes zwischen verbleibenden Bits und eines Abstand zwischen Codevektoren im Codebuch hat, möglich, die Effekte des Codefehlers im Übertragungspfad zu verhindern.
[0405] Mit dem Vektorquantisierungsverfahren des dritten Aspekts der vorliegenden Erfindung ist es durch Verwendung eines durch Gewichtung mit einer zum Definieren des Verzerrungsmaßes verwendeten gewichteten Matrix gefundenen Abstandes als ein Abstand zwischen den Codevektoren möglich, die Effekte des Codefehlers im Übertragungspfad ohne Verursachung einer Charakteristikverschlechterung bei Abwesenheit des Fehlers zu verhindern.
[0406] Bevorzugte Ausführungsformen des oben beschriebenen Vektorquantisierungsverfahrens werden nachfolgend unter Bezugnahme auf die Zeichnungen erläutert.
[0407] Das Vektorquantisierungsverfahren des ersten Aspekts der vorliegenden Erfindung ist ein Vektorquantisierungsverfahren, das die koinzidenten Größenrelationen des Abstandes zwischen Codevektoren im Codebuch und des Hammingabstandes mit dem auf binäre Weise ausgedrückten Index hat und das stark gegen den Übertragungsfehler ist.
[0408] Indessen wird die Erzeugung eines generellen initialen Codebuchs als einer Basis für das oben erwähnte Codebuch erläutert.
[0409] Mit dem oben erwähnten LBG werden die Schwerpunkte in Zellen nur minuziös angeordnet, um optimiert zu werden, doch werden sie in den relativen positionellen Relationen nicht geändert. Deshalb wird die Qualität des auf der Basis des initialen Codebuchs erzeugten Codebuchs unter dem Einfluss des Verfahrens zur Erzeugung des initialen Codebuchs bestimmt. Bei diesem ersten Beispiel wird zur Erzeugung des initialen Codebuchs ein gespaltener Algorithmus bzw. Splittingalgorithmus verwendet.
[0410] Zuerst wird bei der Erzeugung des den Splittingalgorithmus verwendenden initialen Codebuchs der repräsentative Punkt aller Trainingsdaten aus dem Mittel aller Trainingsdaten gefunden. Dann wird dem repräsentativen Punkt ein kleiner Versatz zur Erzeugung zweier repräsentativer Punkte gegeben. Der LBG wird ausgeführt, und dann werden die zwei repräsentativen Punkte mit einem kleinen Versatz in vier repräsentative Punkte geteilt. Wenn die Umwandlung des LBG eine Zahl mal wiederholt wird, wird die Zahl repräsentativer Punkte wie 2, 4, 8, ...,2^n erhöht. Diese Operation wird durch die folgende Formel (55)

$$Y_{(N/2)+i} = \text{modify}\ (y_i, L) \qquad (55)$$

ausgedrückt, wobei $1 \le i \le N/2$ gilt und L das L-te Element anzeigt.
[0411] Demgemäss ist die Erzeugung des den Splittingalgorithmus verwendenden initialen Codebuchs ein Verfahren zur Gewinnung eines N-pegeligen initialen Codebuchs durch die Formel (55) aus dem Codevektor $Y = \{y_1, Y_2, ..., y_{N/2}\}$ eines N/2-pegeligen Vektorquantisierers.
[0412] Auf der rechten Seite der Formel (55) bedeutet „modify" (y_i, L) , dass das L-te Element von (y_1, y_2, ..., y_L, y_k) modifiziert wird und durch (y_1, y_2, ...,$y_L + \varepsilon_0$, y_k) ausgedrückt werden kann. Das heißt, modify (y_i, L) ist eine Funktion zur Verschiebung des L-ten Elements des Codevektors y_i um einen kleinen Betrag ε_0 (oder in anderen Worten ausgedrückt eine addierende Modifikation von $+\varepsilon_0$ zum L-ten Element des Codevektors y_i).
[0413] Dann wird der modifizierte Codevektor $y_L + \varepsilon_0$, als neuer Startcodevektor mit Training durch den LBG verarbeitet und geteilt.
[0414] Bei der Erzeugung des den Splittingalgorithmus verwendenden initialen Codebuchs ist der Euklidabstand um so kleiner je später die Teilung ist. Das erste Beispiel wird durch Verwendung der oben erwähnten Charakteristiken, die nachfolgend unter Bezugnahme auf die **Fig.** 44 erläutert werden, realisiert.
[0415] Die **Fig.** 44 zeigt eine Reihe von Zuständen, bei denen ein einzelner durch Mittelung von Trainingsdaten in einer einzelnen Zelle gefundener repräsentativer Punkt in einer 8-mal geteilten Zelle durch wiederholte Umwandlung des LBG acht repräsentative Punkte wird. Die **Fig.** 44A bis 44D zeigen die Änderung und Richtung der Teilung derart, dass in **Fig.** 44A ein einziger repräsentativer Punkt, in **Fig.** 44B zwei, in **Fig.** 44C vier und in **Fig.** 44D acht vorhanden sind.
[0416] Die Repräsentativpunkte y3 und y_7 in **Fig.** 44D sind durch Teilung von y'_3 in **Fig.** 44C erzeugt. y_3 ist binär ausgedrückt gleich „11", und y_3 und y_7 sind jeweils binär ausgedrückt gleich „011" bzw. „111". Dies zeigt an, dass die Differenz zwischen $y_{(N/2)+i}$ und y_i nur die Polarität (**1** oder **0**) des MBS (oberste Stelle) des Index ist. Demgemäss ist der Abstand zwischen den Codevektoren von $y_{(N/2)+i}$ und y_i sehr kurz. In anderen Worten ausgedrückt wird, wenn die Teilung fortschreitet, der Abstand der Bewegung des Codevektors aufgrund der

Teilung reduziert. Dies bedeutet, dass das korrekte untere Bit auch ein falsches oberes Bit des Index bewältigen kann. Deshalb wird der Effekt des falschen oberen Bits des Index relativ unbedeutend.

[0417] Da es im Sinne einer späteren Verarbeitung zweckdienlich ist, das obere Bit zu betonen bzw. hervorzuheben, werden die MSB und die LSB (unterste Stelle) im Bitarray des binär ausgedrückten Index des Codebuchs zueinander ersetzt. Die Tabelle 1 zeigt die acht Indizes zusammen mit den Codevektoren der **Fig.** 44D, und die Tabelle 2 zeigt die Ersetzung des MSB und LSB miteinander im Bitarray des Index bei konstanten Codevektoren.

TABELLE 1

Index		Codevektor
Binärzahl	Dezimalzahl	
000	0	Y0
001	1	Y1
010	2	Y2
011	3	Y3
100	4	Y4
101	5	Y5
110	6	Y6
111	7	Y7

TABELLE 2

Index		Codevektor
Binärzahl	Dezimalzahl	
000	0	Y0
001	4	Y1
010	2	Y2
011	6	Y3
100	1	Y4
101	5	Y5
110	3	Y6
111	7	Y7

[0418] In der Tabelle 2 korrespondieren die Codevektoren y_3 und y_7 dezimal ausgedrückt mit „6" bzw. „7", und die Codevektoren y_0 und y_4 korrespondieren mit „0" und „1". Die Codevektoren y_3, y_7 und die Codevektoren y_0, y_4 sind, wie aus der **Fig.** 44D hervorgeht, Paare nächster Codevektoren.

[0419] Demgemäss ist die Differenz zwischen „0" und „1" des LSB des binär ausgedrückten Index die Differenz zwischen „0" und 1" 2" und 3" 4" und 6" und 6" und 7". Beispielsweise wird selbst wenn „110" mit „111" verwechselt wird, der Codevektor y_3 nur mit y_7 verwechselt. Auch wird selbst wenn „000" mit „001" verwechselt wird, der Codevektor y_0 mit y_4 verwechselt. Diese Paare Codevektoren sind die Paare nächster Codevektoren in **Fig.** 44D. Kurz: Selbst bei einem Verwechseln auf der LSB-Seite der Indizes ist der Fehler im Abstand von mit den Indizes korrespondierenden Codevektoren klein.

[0420] In den binären Daten des Index ist der Hammingabstand auf der LSB-Seite durch eine koinzidente Größenrelation mit dem Abstand zwischen den Codevektoren gegeben. Demgemäss wird es nur durch Schützen der MSB-Seite der Binärdaten des Index alleine mit dem Fehlerkorrekturcode möglich, den Effekt des Fehlers im Übertragungspfad auf das Minimum zu steuern.

[0421] Als nächstes wird ein Beispiel des Vektorquantisierungsverfahrens des zweiten Aspekts der vorliegenden Erfindung erläutert.

[0422] Das Vektorquantisierungsverfahren des zweiten Aspekts der vorliegenden Erfindung ist ein Verfahren,

bei welchem der Hammingabstand zum Zeitpunkt des Trainings des Vektorquantisierers in Rechnung gestellt wird.

[0423] Zuerst wird vor der Erläuterung des Vektorquantisierungsverfahrens des zweiten Aspekts ein Vektorquantisierungsverfahren, bei dem der Vektorquantisierer an einen Kommunikationspfad angepasst ist und bei dem ein in **Fig.** 45 gezeigtes Kommunikationssystem hinsichtlich Kommunikationsfehlern verwendet ist, wodurch eine Verschlechterung von Charakteristiken bei der Abwesenheit von Fehlern verursacht wird, erläutert.

[0424] Bei dem in **Fig.** 45 gezeigten Kommunikationssystem wird ein von einen Eingangsanschluss **821** in einen Vektorquantisierer **822** eingegebener Eingangsvektor X durch Abbildung in einen Abbildungsabschnitt **822a** verarbeitet, um y_i auszugeben. Der Index i wird von einem Codierer **822b** über einen Kommunikationspfad **823** zu einem Dekodierer **824** als Binärdaten übertragen. Der Dekodierer **824** inversquantisiert den übertragenen Index und gibt Daten von einem Ausgangsanschluss **825** ab. Es sei angenommen, dass die Wahrscheinlichkeit, dass sich der Index i während des Zeitpunkts, bei dem durch das Kommunikationssystem **823** ein Fehler zum Index i addiert wird, und bei dem der Index i mit dem Fehler zum Dekodierer **824** übertragen wird, die Wahrscheinlichkeit P(j|i) ist. Das heißt, die Wahrscheinlichkeit P(j|i) ist die Wahrscheinlichkeit, dass der Übertragungsindex i als der Empfangsindex j empfangen wird. In einem symmetrischen binären Kommunikationspfad (Binärdaten-Kommunikationspfad) in welchem die Bitfehlerrate e ist, kann die Wahrscheinlichkeit P(j|i) durch

$$P(j|i) = e^{d_{ij}}(1-e)^{s-d_{ij}} \qquad (56)$$

ausgedrückt werden, wobei d_{ij} den binär ausgedrückten Hammingabstand mit dem Übertragungsindex i und dem Empfangsindex j anzeigt und S die binär ausgedrückte Zahl von Stellen (Bitzahl) mit dem Übertragungsindex i und dem Empfangsindex j ausdrückt.

[0425] Unter der Bedingung, dass der Kommunikationspfadfehler mit der durch die Formel (56) gezeigten Wahrscheinlichkeit P(j|i) erzeugt wird, ist der optimale Flächenschwerpunkt (Repräsentativer Punkt) y_u zu dem Zeitpunkt, bei dem die Zellteilung $\{R_i\}$ bereitgestellt ist, wie folgt ausgedrückt:

$$y_u = \frac{\sum_{i=1}^{N} P(u|i) \sum_{j:X_{Ri}} X_j}{\sum_{i=1}^{N} P(u|i)|R_i|} \qquad (57)$$

[0426] In der Formel (57) bezeichnet $|R_i|$ die Zahl Trainingsvektoren im partiellen Raum R_i. Normalerweise ist ein repräsentativer Punkt das durch die Summe von Trainingsvektoren X im partiellen Raum geteilt durch die Zahl der Trainingsvektoren X gefundene Mittel. Bei der Formel (57) jedoch wird das gewichtete Mittel gefunden, das durch Gewichtung der Summe des Mittels der Trainingsvektoren X in allen partiellen Räumen mit der Fehlerwahrscheinlichkeit P(u|i) erzeugt wird. Gemäss der Formel (57) kann davon gesprochen werden, das gewichtete Mittel in dem mit der Wahrscheinlichkeit des sich in den Empfangsindex u sich ändernden Übertragungsindex i gewichteten Flächenschwerpunkt auszudrücken.

[0427] Die optimale Teilung R_u eines Codebuchs {yi : i = 1, 2, ..., N)} kann durch die folgende Formel

$$R_u = \left\{ X : \sum_{j=1}^{N} P(j|u) d(X, y_j) \le \sum_{j=1}^{N} P(j|i) d(X, y_j) \; \text{fürallei} \neq u \right\} \qquad (58)$$

ausgedrückt werden. Kurz ausgedrückt drückt die Formel (58) einen partiellen Raum aus, der durch einen Satz Eingangsvektoren X gebildet ist, der einen Index u mit dem minimalen gewichteten Mittel von Verzerrungsmaßen d(X, yj), der mit der Wahrscheinlichkeit, dass sich der vom Codierer ausgegebene Index u im Übertragungspfad in j ändert, wählt. Zu diesem Zeitpunkt kann die optimale Teilungsbedingung wie folgt ausgedrückt werden:

$$C(X) = Uiff \sum_{j=1}^{N} P(j|u) d(X, y_j) \le \sum_{j=1}^{N} P(j|i) d(X, y_j) \; \text{fürallei} \in I \qquad (59$$

[0428] Das optimale Codebuch für die Bitfehlerrate wird wie oben beschrieben erzeugt. Da jedoch dies ein hinsichtlich der Bitfehlerrate erzeugtes Codebuch ist, werden Charakteristiken bei Abwesenheit des Fehlers mehr als bei dem konventionellen Vektorquantisierungsverfahren verschlechtert.

[0429] Infolgedessen hat der Erfinder der vorliegenden Erfindung ein Vektorquantisierungsverfahren als die zweite Ausführungsform des Vektorquantisierungsverfahrens in Betracht gezogen, das beim Training des Vektorquantisierers den Hammingabstand in Rechnung stellt und keine Verschlechterung von Charakteristiken bei Abwesenheit des Fehlers verursacht.

[0430] Speziell wird die Bitfehlerrate e auf 0,5 gesetzt, ein im Kommunikationspfad nicht zuverlässiger Wert. Kurz ausgedrückt werden sowohl P (u|i) als auch P (i|u) konstant eingestellt. Dies erzeugt einen instabilen Zustand, in welchem unbekannt ist, wohin die Zelle bewegt wird. Zur Vermeidung dieses instabilen Zustandes wird am meisten bevorzugt, den Mittelpunkt der Zelle auf der Decodiererseite auszugeben. Dies bedeutet, dass bei der Formel (57) y_u auf einen einzelnen Punkt (den Schwerpunkt des ganzen Trainingssatzes) konzentriert ist. Auf der Codiererseite werden alle Eingangsvektoren X mit einer Abbildung auf den gleichen Codevektor verarbeitet, wie es durch die Formel (59) gezeigt ist. Kurz ausgedrückt ist das Codebuch in einem Zustand eines hohen Energiepegels für jede Übertragung.

[0431] Wenn die Bitfehlerrate e graduell von 0,5 auf 0 reduziert wird, wodurch die Struktur graduell fixiert wird, um die Bitfehlerrate letztendlich auf 0 zu reduzieren, kann ein partieller Raum derart, dass er die ganzen Basistrainingsdaten X abdeckt, erzeugt werden. Das heißt, die Wirkung des Hammingabstandes der Indizes der benachbarten Zellen im LBG-Trainingsprozess wird durch B(i|j) reflektiert. Insbesondere bei dem durch die Formel (57) angezeigten repräsentativen Punkt wird dessen Aktualisierung durch den repräsentativen Punkt einer anderen Zelle während der Ausführung einer Gewichtung entsprechend dem Hammingabstand beeinflusst. Auf diese Weise korrespondiert der Prozess der graduellen Reduzierung der Fehlerrate von 0,5 auf 0 mit einem Prozess einer Kühlung durch graduelle Entfernung von Wärme.

[0432] Bei dieser Stufe wird ein Verarbeitungsfluss des oben erwähnten zweiten Beispiels, das heißt das Vektorquantisierungsverfahren, das keine Verschlechterung von Charakteristiken auch bei der Abwesenheit des Fehlers verursacht, wobei der Hammingabstand zum Zeitpunkt des Trainings der Vektorquantisierung in Rechnung gestellt wird, unter Bezugnahme auf die **Fig.** 46 erläutert.

[0433] Zuerst wird beim Schritt 5811 eine Initialisierung ausgeführt. Speziell wird die Verzerrung D-1 auf Unendlich eingestellt, und die Wiederholungszahl n wird auf „0" (n = 0) gesetzt, während die Bitfehlerrate e auf 0,49 gesetzt wird. Auch werden Y_0 ε und n_m als das initiale Codebuch, die Schwelle bzw. die maximale Iterationszahl definiert.

[0434] Beim Schritt 5812 werden mit dem beim Schritt S811 gegebenen initialen Codebuch X_0 alle bei diesem Zustand vorhandenen Trainingsdaten unter der Nächstnachbarbedingung kodiert. Kurz ausgedrückt wird das initiale Codebuch durch Abbildung verarbeitet.

[0435] Beim Schritt 5813 wird eine Verzerrungsberechnung zur Berechnung der Quadratsumme des Abstandes zwischen den Eingangsdaten und den Ausgangsdaten ausgeführt.

[0436] Beim Schritt 5814 wird festgestellt, ob die Reduktionsrate der aus der vorhergehenden Verzerrung D_{-1} und der gegenwärtigen Verzerrung D_n beim Schritt 5813 kleiner als die Schwelle ε wird oder nicht, oder ob die Iterationszahl n die maximale Iterationszahl n_m, die im Voraus bestimmt ist, erreicht hat. Wenn JA gewählt ist, geht die Operation zum Schritt 5815 vor, und wenn NEIN gewählt ist, geht die Operation zum Schritt 5816 vor.

[0437] Beim Schritt 5815 wird festgestellt, ob die Bitfehlerrate e gleich 0 wird nicht. Wenn JA gewählt ist, endet der Operationsfluss, und wenn NEIN gewählt ist, geht die Operation zum Schritt S819 vor.

[0438] Der Schritt S816 dient dazu, zu vermeiden, dass der Codevektor mit den Eingangsdaten nicht als Ganzes mit der Abbildung verarbeitet wird, die vorhanden ist, wenn beim Schritt 5811 ein unrichtiges initiales Codebuch vorhanden ist. Normalerweise wird der Codevektor mit den nicht durch Abbildung verarbeiteten Eingangsdaten in die Nähe einer Zelle mit der größten Verzerrung verschoben.

[0439] Beim Schritt S817 wird ein neuer Schwerpunkt durch Berechnung auf der Basis der Formel (57) gefunden.

[0440] Die zum Schritt 5818 vorgehende Operation kehrt zum Schritt S812 zurück, und dieser Operationsfluss wird wiederholt, bis beim Schritt S815 JA gewählt ist.

[0441] Beim Schritt S819 wird α (beispielsweise α = 0,01) von der Bitfehlerrate e für jeden Fluss reduziert, bis beim Schritt 5815 die Entscheidung über die Bitfehlerrate e = 0 getroffen wird.

[0442] Bei der vorliegenden zweiten Ausführungsform kann das optimierte Codebuch schließlich durch den oben erwähnten Operationsfluss mit der Fehlerrate e = 0 erzeugt werden, und es wird eine kleine Verschlechterung von Vektorquantisierungscharakteristiken bei der Abwesenheit des Fehlers erzeugt.

[0443] Auch wenn ein oberes Bit g mit einer Fehlerkorrektur geschützt wird, während ein unteres Bit W-g nicht mit der Fehlerkorrektur in einem durch W Bits ausgedrückten Index verarbeitet wird, kann P(i|j) durch Reflektieren nur des Hammingabstandes des unteren Bits W-g mit der Formel (56) gefunden werden. Das heißt, wenn der Index die gleichen oberen g Bits aufweist, wird der Hammingabstand betrachtet. Wenn es auch nur ein einzelnes verschiedenes Bit unter den oberen g Bits gibt, wird der Index auf P(i|j) = 0 gesetzt. Kurz ausgedrückt wird angenommen, dass das obere g-Bit, das mit der Fehlerkorrektur gestützt ist, fehlerfrei ist.

[0444] Als nächstes wird das dritte Beispiel des Vektorquantisierungsverfahrens des dritten Aspekts der vorliegenden Erfindung erläutert.

[0445] Beim dritten Beispiel des Vektorquantisierungsverfahrens ist ein initiales N-Punkt-Codebuch mit einer gewünschten Struktur vorgesehen. Wenn ein initiales Codebuch, das eine analoge Relation zwischen dem Hammingabstand und dem Euklidabstand aufweist, kollabiert die Struktur nicht, selbst wenn sie durch den konventionellen LBG trainiert wird.
[0446] Bei der Erzeugung des initialen Codebuchs bei diesem dritten Beispiel wird der Repräsentativpunkt jedes Mal bei Eingabe eines einzelnen Abtastwertes der Trainingsdaten aktualisiert. Normalerweise ist, wie in der **Fig.** 47 gezeigt der durch die Eingangstrainingsdaten X in einer Zelle m_j aktualisierte Repräsentativpunkt nur $m_j \cdot m_j$ neu beispielsweise m_{j+1} und m_{j+2} werden wie folgt aktualisiert:

$$m_j \text{ neu} = m_{j\,alt} + \Delta m_j \tag{60}$$

wobei $\Delta m_j = (X - m_{j\,alt}) \cdot \alpha$ $\alpha < 1$ gilt.
[0447] Kurz ausgedrückt wird die Abtastung mit allen Trainingsdaten X ausgeführt. Dann wird die gleiche Abtastung mit verkleinertem α ausgeführt. Schließlich wird bei weiterer Reduzierung von α eine Umwandlung bis 0 ausgeführt, wodurch das initiale Codebuch erzeugt wird.
[0448] Bei diesem dritten Beispiel werden die Eingangstrainingsdaten X nicht nur bei m_j reflektiert, sondern auch bei m_{j+1} und m_{j+2}, so dass alle peripheren Zellen beeinflusst werden. Beispielsweise im Fall von m_{j+1} wird m_{j+1} neu wie folgt:

$$m_{j+1} \text{ neu} = m_{j+1} \text{ alt} + \Delta m_{j+1} \text{ wobei } \Delta m_{j+1} = (X - m_{j+1\,alt}) \cdot \alpha \cdot f(j+1, j) \; \alpha < 1 \text{ gilt.} \tag{61}$$

[0449] In der Formel (61) ist f(j + 1, j) eine Funktion zum Zurückbringen eines mit dem Kehrwert des Hammingabstandes von j und j + 1 proportionaler Werts, beispielsweise f(j + 1, j) = P(j + 1|j).
[0450] Eine generellere Form der Formel (61) ist folgende:

$$m_j \text{ neu} = m_{j\,alt} + \Delta m_j \text{ wobei } \Delta m_j = (X - m_j) \cdot \alpha \cdot f(j, C(X)) \; \alpha < 1 \text{ gilt.} \tag{62}$$

[0451] C(X) in der Formel (62) bringt einen Index u einer Zelle mit dem Schwerpunkt nächst dem Eingangssignal X zurück. C(X) kann wie folgt definiert werden:

$$C(X) \; U \text{ iff } d(X, y_u) \leq d(X, y_j) \text{ für alle } i \in I. \tag{63}$$

[0452] Als ein Beispiel der Funktion f kann

$$f(j, C(X)) = P(j|C(X))$$

verwendet werden. Infolgedessen wird bei der dritten Ausführungsform das initiale Codebuch durch das oben beschriebene Aktualisierungsverfahren erzeugt, und dann wird der LBG ausgeführt.
[0453] Demgemäss kollabiert bei der dritten Ausführungsform der vorliegenden Erfindung, wenn das initiale N-Punkt-Codebuch mit der analogen Relation zwischen dem Hammingabstand und dem Euclidabstand erzeugt wird, die Struktur nicht, selbst wenn das Training mit dem konventionellen LBG ausgeführt wird.
[0454] Gemäß dem wie oben beschriebenen Vektorquantisierungsverfahren werden der Abstand von Codevektoren in dem aus mehreren M-dimensionalen Codevektoren mit M Einheiten aus Daten als M-dimensionale Vektoren bestehenden Codebuch und der Ham mingabstand zum Zeitpunkt des Ausdruckens der Indizes der gesuchten Codevektoren in der binären Weise in der Größe koinzident gemacht. Auch wird ein Teil von Bits der die Indizes der gesuchten Vektoren ausdrückenden binären Daten mit dem Fehlerkorrekturcode geschützt, während der Hammingabstand der verbleibenden Bits und der Abstand zwischen den Codevektoren in dem Codebuch in der Größe koinzident gemacht werden. Auf diese Weise ist es möglich, den Effekt des Codefehlers im Übertragungspfad zu steuern. Außerdem ist es durch Einstellen des durch Gewichtung mit der zum Definieren des Verzerrungsmaßes gefundenen Abstandes als der Abstand zwischen den Vektoren möglich, die Wirkung des Codefehlers im Übertragungspfad zu steuern, ohne dass eine Verschlechterung von Charakteristiken bei der Abwesenheit des Fehlers verursacht wird.
[0455] Als nächstes wird eine Anwendung des Sprach-Analyse-Synthese-Verfahrens auf die Stimmensignal-Analyse-Synthese-Kodierungseinrichtung erläutert.
[0456] Bei dem in der Stimmen-Analyse-Synthese-Einrichtung angewendeten Stimmen-Analyse-Synthese-Verfahren ist es notwendig, die Phase auf der Rnalyseseite an die Phase auf der Syntheseseite anzupassen. In diesem Fall kann eine lineare Vorhersage durch die Winkelfrequenz und eine Modifikation durch das Weißrauschen zur Gewinnung von Phaseninformation auf der Syntheseseite verwendet werden. Jedoch ist es nicht möglich, mit dem Weißrauschen eine Steuerung von Rauschen oder Fehlern durch den realen Wert der Phase und der Vorhersage auszuführen.

[0457] Auch wird der Pegel des Weißrauschens bei einem Verhältnis stimmloser Töne im ganzen Band geändert, so dass er in dem Modifikationstherm zu verwenden ist. Deshalb kann in dem Fall, dass ein großes Verhältnis von stimmhaften Tönen enthaltende Blöcke aufeinanderfolgend existieren die Modifikation nicht nur durch Vorhersage ausgeführt werden. Dies hat zur Folge, dass, wenn starke Vokale sich lange fortsetzen, Fehler akkumuliert werden, was die Tonqualität verschlechtert.

[0458] Infolgedessen wird ein Sprach-Analyse-Synthese-Verfahren vorgeschlagen, durch das eine Verbesserung der Tonqualität durch Verwendung von Rauschen, das die Größe und Diffusion für eine Modifikation aufgrund einer Vorhersage steuern kann, realisiert werden kann.

[0459] Das heißt, das Sprach-Rnalyse-Synthese-Verfahren weist die Schritte auf: Teilen eines Spracheingangssignals auf der blockweisen Basis und Finden von Tonhöhendaten im Block, Umwandeln des Sprachsignals auf der blockweisen Basis in das Signal auf der Frequenzachse und Finden von Daten auf der Frequenzachse, Teilen der Daten auf der Frequenzachse in mehrere Bänder auf der Basis der Tonhöhendaten, Finden von Leistungsinformation für jedes der geteilten Bänder und Feststellen von Information darüber, ob das Band stimmhaft oder stimmlos ist, Übertragen der bei dem obigen Prozessen gefundenen Tonhöhendaten, der Leistungsinformation für jedes Band und der Stimmhaft/Stimmlos-Entscheidungsinformation, Vorhersagen einer Blockende-Randphase auf der Basis der durch Übertragung erhaltenen Tonhöhendaten für jeden Block und einer initialen Blockphase, und Modifizieren der vorhergesagten Blockende-Randphase unter Verwendung eines eine Diffusion entsprechend jedem Band aufweisenden Rauschens. Vorzugsweise ist das oben erwähnte Rauschen ein Gaußsches Rauschen.

[0460] Gemäß einem solchen Sprach-Analyse-Synthese-Verfahren werden die Leistungsinformation und die Stimmhaft/Stimmlos-Entscheidungsinformation auf der Analyseseite gefunden und dann für jedes der mehreren Bänder, die durch Teilen der durch Umwandlung des blockweisen Sprachsignals in das Signal auf der Frequenzachse auf der Basis der aus dem blockweisen Sprachsignal gefundenen Tonhöhendaten erhaltenen Daten auf der Frequenzachse erzeugt sind, übertragen, und die Blockende-Randphase wird auf der Syntheseseite auf der Basis der Tonhöhendaten für jeden durch Übertragung erhaltenen Block und der initialen Blockphase vorhergesagt. Dann wird die vorhergesagte Ende-Randphase unter Verwendung des Gaußschen Rauschens mit einer Diffusion entsprechend jedem Band modifiziert. Auf diese Weise ist es möglich, einen Fehler oder eine Differenz zwischen dem vorhergesagten Phasenwert und dem realen Wert zu steuern.

[0461] Ein konkretes Beispiel, bei welchem das oben beschriebene Sprach-Analyse-Synthese-Verfahren auf die Sprachsignal-Analyse-Synthese-Kodierungseinrichtung (den sogenannten Vocoder) angewendet ist, wird unter Bezugnahme auf die Zeichnungen erläutert. Die Analyse-Synthese-Kodierungseinrichtung führt eine Modellierung derart aus, dass ein stimmhafter Abschnitt und ein stimmloser Abschnitt in einem koinzidenten Frequenzachsenbereich (im gleichen Block oder gleichen Rahmen) vorhanden sind.

[0462] Die **Fig.** 48 ist eine schematische Darstellung, die eine schematische Anordnung eines ganzen Beispiels zeigt, bei welchem das Sprach-Analyse-Synthese-Verfahren auf die Sprachsigna1-Analyse-Synthese-Kodierungseinrichtung angewendet ist.

[0463] Bei der **Fig.** 48 weist die Sprach-Analyse-Synthese-Kodierungseinrichtung auf: einen Analyseabschnitt 910 zum Analysieren von Tonhöhendaten usw. aus einem Spracheingangssignal und einen Syntheseabschnitt 920 zum Empfang verschiedener Informationstypen wie beispielsweise die vom Analyseabschnitt **910** durch einen Übertragungsabschnitt **902** übertragenen Tonhöhendaten, Synthetisieren stimmhafter bzw. stimmloser Töne und Synthetisieren der stimmhaften und stimmlosen Töne zusammen.

[0464] Der Analyseabschnitt **910** weist auf: einen Blockextraktionsabschnitt 911 zum Ausgeben eines von einem Eingangsanschluss **901** auf der blockweisen Basis eingegebenen Sprachsignals, wobei jeder Block aus einer vorbestimmten Zahl Abtastwerten (N Abtastwerte) besteht, einen Tonhöhendatenextraktionsabschnitt 912 zum Extrahieren von Tonhöhendaten aus dem Eingangsprachsignal auf der blockweisen Basis aus dem Blockextraktionsabschnitt **911**, einen Datenumwandlungsabschnitt 913 zum Finden von auf der Frequenzachse umgewandelten Daten aus dem Eingangssprachsignal auf der blockweisen Basis aus dem Blockextraktionsabschnitt **911**, einen Bandteilungsabschnitt 914 zum Teilen der Daten auf der Frequenzachse aus dem Datenumwandlungsabschnitt **913** in mehrere Bänder auf der Basis der Tonhöhendaten des Tonhöhendatenextraktionsabschnitts **912**, und einen Amplitudendaten- und V/UV-Entscheidungsinformations-Detektionsabschnitt 915 zum Finden von Leistungsinformation (Amplitudeninformation) für jedes Band des Bandteilungsabschnitts **914** und einer Entscheidungsinformation darüber, ob das Band stimmhaft (V) oder stimmlos (W) ist.

[0465] Der Syntheseabschnitt **920** empfängt die Tonhöhendaten, die V/UV-Entscheidungsinformation und die Amplitudeninformation, die vom Übertragungsabschnitt **902** übertragen werden, aus dem Analyseabschnitt **910**. Dann synthetisiert der Syntheseabschnitt **920** den stimmhaften Ton durch einen Stimmhafttonsyntheseabschnitt **921** und den stimmlosen Ton durch einen 5timmlostonsyntheseabschnitt 927 und addiert den synthetisierten stimmhaften und stimmlosen Ton durch einen Addierer **928** zusammen. Dann gibt der Syntheseabschnitt **920** das synthetisierte Sprachsignal aus dem Ausgangsanschluss **903** aus.

[0466] Die oben erwähnte Information wird durch Verarbeitung der Daten in dem Block aus den N Abtastwerten, beispielsweise 256 Abtastwerte, erhalten. Da jedoch der Block auf der Basis eines Rahmens aus L Ab-

tastwerten als eine Einheit auf der Zeitachse vorwärtsgeht, werden die übertragenen Daten auf der rahmenweisen Basis erhalten. Das heißt, die Tonhöhendaten, die V/UV-Information und die Amplitudeninformation werden mit dem Rahmenzyklus aktualisiert. Der Stimmhafttonsyntheseabschnitt **921** weist auf: einen Phasenvorhersageabschnitt 922 zum Vorhersagen einer Rahmenende-Randphase (Startrandphase des nächsten Syntheserahmens) auf der Basis der Tonhöhendaten und einer initialen Rahmenphase, die von einem Eingangsanschluss **904** zugeführt sind, einen Phasenmodifikationsabschnitt 924 zum Modifizieren der Vorhersage aus dem Phasenvorhersageabschnitt **922** unter Verwendung eines Modifikationsterms aus einem Rauschenadditionsabschnitt 923, dem die Phasendaten und die V/UV-Entscheidungsinformation zugeführt sind, einen Sinuswellenerzeugungsabschnitt 925 zum Auslesen und Ausgeben einer Sinuswelle aus einem nicht gezeigten Sinuswellen-ROM auf der Basis der Modifikationsphaseninformation aus dem Phasenmodifikationsabschnitt **924**, und einen Amplitudenverstärkungsabschnitt 926, dem die Amplitudeninformation zum Verstärken der Amplitude der Sinuswelle aus dem Sinuswellenerzeugungsabschnitt **925** zugeführt ist.

[0467] Die Tonhöhendaten, die V/UV-Entscheidungsinformation und die Amplitudeninformation werden dem Stimmlostonsyntheseabschnitt **927** zugeführt, bei dem beispielsweise das Weißrauschen durch Filterung mit einem nicht gezeigten Bandpassfilter verarbeitet wird, um eine Stimmlostonwellenform auf der Zeitachse zu synthetisieren.

[0468] Der Addierer **928** addiert den vom Stimmhafttonsyntheseabschnitt **921** und Stimmlostonsyntheseabschnitt **927** synthetisierten stimmhaften Ton bzw, stimmlosen Ton mit einem festen Mischungsverhältnis. Das addierte Sprachsignal wird aus dem Ausgangsanschluss **903** als das Sprachsignal ausgegeben.

[0469] Im Phasenvorhersageabschnitt 922 im Stimmhafttonsyntheseabschnitt **921** des Syntheseabschnitts **920** wird unter der Annahme, dass die Phase (initiale Rahmenphase) der m-ten Oberwelle zum Zeitpunkt **0** (Kopf des Rahmens) gleich ψ_{Om} ist, die Phase ψ_{Lm} ab Ende des Rahmens wie folgt vorhergesagt:

$$\psi_{Lm} = \psi_{Om} + m(\omega_{01} + \omega_{L1})L/2 \quad (64)$$

[0470] Die Phase jedes Bandes ϕm wird wie folgt gefunden:

$$\Phi_m = \psi_{Lm} + \varepsilon_m \quad (65)$$

[0471] In den Formeln (64) und (65) bezeichnet ω_{O1} die fundamentale Winkelfrequenz am Startrand (n = 0) des Syntheserahmens, und ω_{L1} bezeichnet die fundamentale Winkelfrequenz am Endrand des Syntheserahmens (n = L, Startrand des nächsten Syntheserahmens), während ε_m den Vorhersagemodifikationsterm in jedem Band bezeichnet.

[0472] Durch die Formel (64) findet der Phasenvorhersageabschnitt **922** eine Phase als die Vorhersagephase zum Zeitpunkt L durch Multiplizieren der mittleren Winkelfrequenz der m-ten Oberschwingung bzw. Oberwelle mit der Zeit und durch Hinzuaddieren der initialen Phase der m-ten Oberwelle. Aus der Formel (65) wird gefunden, dass die Phase ε_m jedes Bandes ein durch Addieren des Vorhersagemodifikationsterms ε_m zur Vorhersagephase erzeugter Wert ist.

[0473] Für den Vorhersagemodifikationsterms ε_m kann wegen seiner zufälligen Verteilung zwischen den Bändern eine Zufallszahl verwendet werden. Jedoch wird bei der vorliegenden Ausführungsform ein Gaußsches Rauschen verwendet. Das Gaußsche Rauschen ist ein Rauschen, dessen Diffusion, wie in **Fig.** 49 gezeigt, in Richtung zum höheren Frequenzband zunimmt (beispielsweise von ε_1 auf ε_{10}. Das Gaußsche Rauschen approximiert den Vorhersagewert der Phase richtig auf den realen Wert der Phase.

[0474] Wenn wie in **Fig.** 49 gezeigt die Diffusion einfach proportional zu m ist, wird der Vorhersagemodifikationsterm ε_m durch

$$\varepsilon_m = h_1 N(O, k_i) \quad (66)$$

angezeigt, wobei h_1, k_i und 0 eine Konstante, einen Bruchteil bzw. einen Mittelwert bedeuten.

[0475] Wenn das ganze Band in zwei Bänder aus einem stimmhaften Band und einem stimmlosen Band geteilt wird, wobei der stimmlose Abschnitt größer ist, werden die Phasen von die Stimme bzw. Sprache bildenden Frequenzkomponenten zufälliger. Deshalb kann der Vorhersagemodifikationsterm ε_m ausgedrückt werden durch

$$\varepsilon_m = h_2 n_{uj} N(O, k_i) \quad (67)$$

wobei h_2, k_i, 0 und n_{uj} eine Konstante, einen Bruch, ein Mittel bzw, die Zahl stimmloser Bänder in einem Block j bedeuten.

[0476] Wenn es wie oben beschrieben keine zufällige Verteilung zwischen Bändern gibt, insbesondere aufgrund von lange fortgesetzten Vokalen oder wenn Vokale auf Konsonanten und stimmlose Töne verschoben

werden, verschlechtert der in den Formeln (66) und (67) gezeigte Vorhersagemodifikationsterm eher die Qualität des synthetischen Tons. Deshalb wird bei Zulässigkeit einer Verzögerung der Amplitudeninformations(Leistungs)-S-Pegel eines vorhergehenden Rahmens oder eine Reduktion des stimmhaften Tonabschnitts geprüft, wobei der Modifikationsterm ε_m durch

$$\varepsilon_m = h_3 \max (a, S_j - S_{j+1}) N(O, ki) \quad (68)$$

$$\varepsilon_m = h_4 \max (b, n_{vj} - n_{v(j+1)}) N(O, k_i) \quad (69)$$

eingestellt, wobei a, b, h_3 und h_4 Konstanten sind.

[0477] Wenn außerdem die Tonhöhendaten beim Tonhöhendatenextraktionsabschnitt **912** niedrig sind, wird die Zahl der Frequenzbänder erhöht, und es wird der umgekehrte Effekt der Ausrichtung der Phasen erhöht. Bei in Betracht ziehen dieses wird der Modifikationsterm ε_m ausgedrückt durch

$$\varepsilon_m = f(S_j, h_j) N(O, k_i) \quad (70)$$

wobei f die Frequenz bedeutet.

[0478] Bei der die vorliegende Erfindung auf die Sprachsignal-Analyse-Synthese-Kodierungseinrichtung anwendenden Ausführungsform können die Größe und die Diffusion des für die Phasenvorhersagemodifikation verwendenden Rauschens durch Verwendung eines Gaußschen Rauschens gesteuert werden.

[0479] Bei dem Beispiel, bei dem ein solches Sprach-Analyse-Synthese-Verfahren auf den bezüglich der **Fig.** 1 bis 7 erläuterten MBE angewendet ist, können die Größe und Diffusion des für die Phasenvorhersage verwendeten Rauschens durch Verwendung eines Gaußschen Rauschens gesteuert werden.

[0480] Bei den oben beschriebenen Sprach-Analyse-Synthese-Verfahren werden die Leistungsinformation und die V/UV-Entscheidungsinformation auf der Analyseseite gefunden und für jedes der mehreren Frequenzbänder, die durch Teilen der durch Umwandeln des blockweisen Sprachsignals in das Signal auf der Frequenzachse auf der Basis der aus dem blockweisen Sprachsignal gefundenen Tonhöhendaten erhaltenen Frequenzachsendaten erzeugt werden, übertragen, und die Blockende-Randphase wird auf der Syntheseseite auf der Basis der für jeden Block durch Übertragung erhaltenen Tonhöhendaten und der initialen Blockphase vorhergesagt. Dann wird die vorhergesagte Ende-Randphase unter Verwendung des eine Diffusion entsprechend jedem Band aufweisenden Gaußschen Rauschens modifiziert. Auf diese Weise ist es möglich, die Größe und die Diffusion des Rauschens zu steuern und infolgedessen eine Verbesserung in der Tonqualität zu erwarten. Auch ist es durch Verwendung des Signalpegels der Stimme bzw. Sprache und deren zeitlichen Änderungen möglich, eine Akkumulation von Fehlern zu verhindern und eine Verschlechterung der Tonqualität in einem Vokalabschnitt oder bei einem Umschaltpunkt von dem Vokalabschnitt auf einen Konsonantenabschnitt zu verhindern.

[0481] Indessen ist die vorliegende Erfindung nicht auf die obigen Ausführungsformen beschränkt. Beispielsweise kann als Eingangssignal nicht nur das Sprachsignal sondern auch ein akustisches Signal verwendet werden. Der Charakteristiken des Eingangsaudiosignals (Sprachsignal oder akustisches Signal) ausdrückende Parameter ist nicht auf die V/UV-Entscheidungsinformation beschränkt, sondern es können der Tonhöhenwert, die Stärke von Tonhöhenkomponenten, die Neigung und der Pegel des Signalspektrums usw. verwendet werden. Außerdem kann anstelle dieser charakteristischen Parameter ein Teil der entsprechend den Codierungsverfahren ursprünglich zu übertragenden Parameterinformation verwendet werden. Auch können die Charakteristikparameter separat übertragen werden. Im Fall der Verwendung anderer Übertragungsparameter können diese Parameter als ein adaptives Codebuch betrachtet werden, und in dem Fall der separaten Übertragung der Charakteristikparameter können die Parameter als ein strukturiertes Codebuch betrachtet werden.

Patentansprüche

1. Hocheffizientes Codierungsverfahren, das die Schritte aufweist:

Finden von Interblockdifferenzdaten als einen M-dimensionalen Vektor wobei M eine ganze Zahl größer als eins ist, durch Nehmen (**614**) einer Interblockdifferenz von durch Teilen (**612a**) eines Eingangsaudiosignals in Blöcke und Umwandeln (**612b**) der resultierenden Blocksignale in Signale auf einer Frequenzachse erhaltenen Daten, und

Verarbeiten (**615**) der Interblockdifferenzdaten des Mdimensionalen Vektors durch Vektorquantisierung.

Es folgen 37 Blatt Zeichnungen

Anhängende Zeichnungen

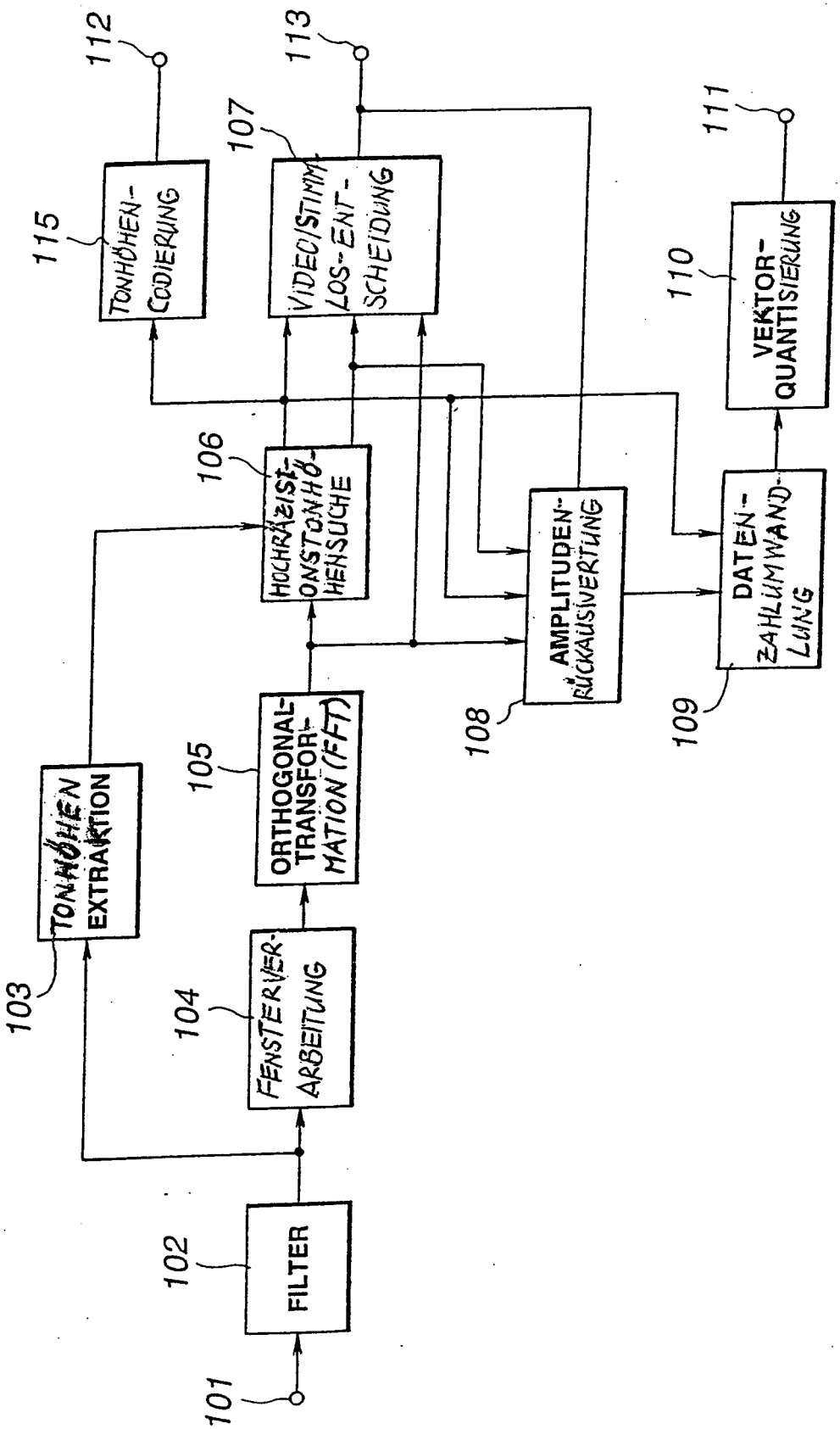


FIG.1

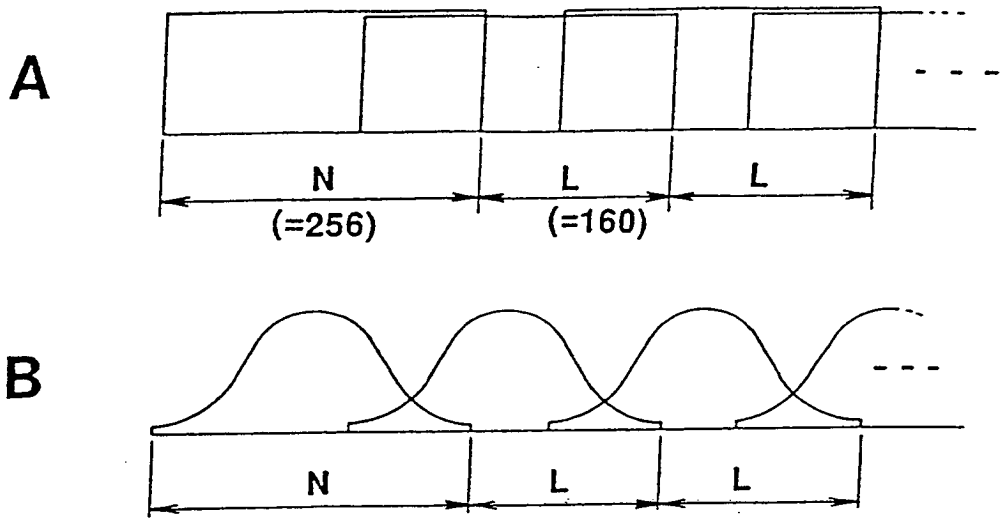


FIG.2

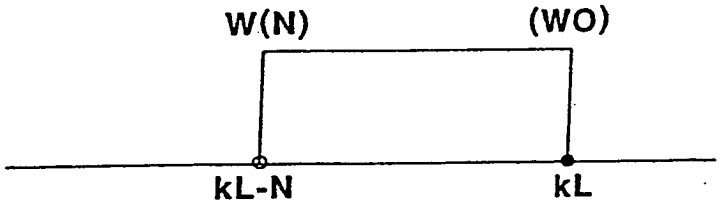


FIG.3

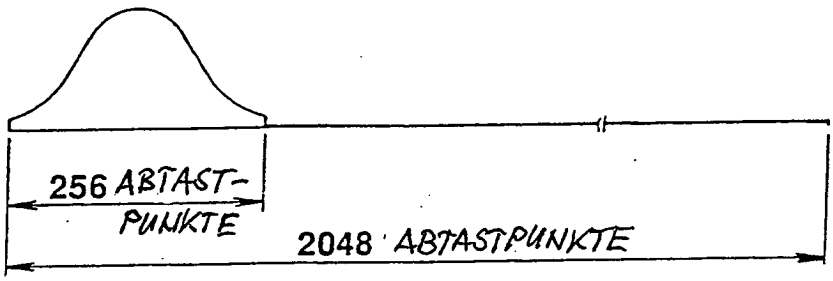


FIG.4

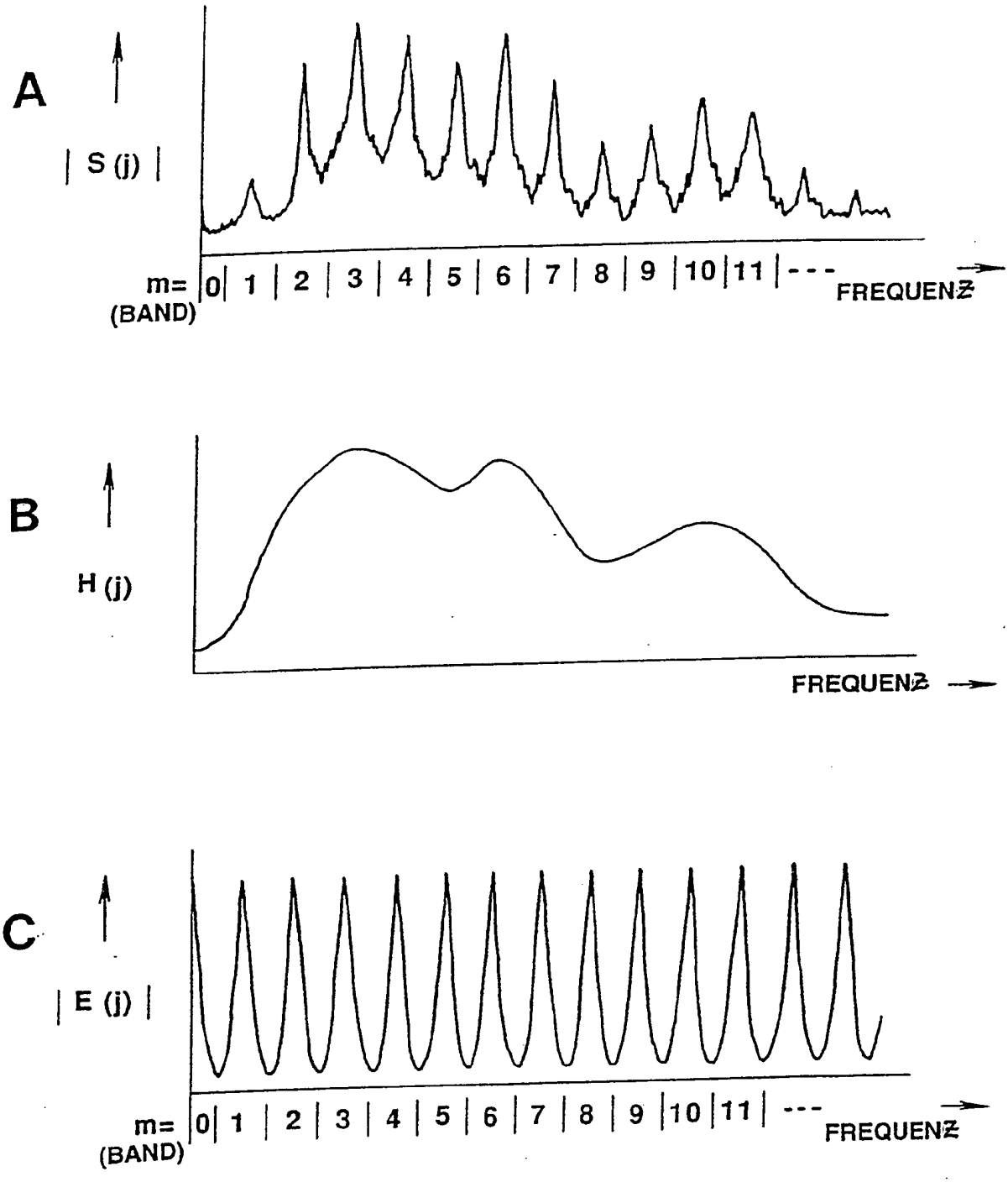

A
| S (j) |
m=
(BAND)
0 1 2 3 4 5 6 7 8 9 10 11 ---
FREQUENZ
B
H (j)
FREQUENZ
C
| E (j) |
m=
(BAND)
0 1 2 3 4 5 6 7 8 9 10 11 ---
FREQUENZ

FIG.5

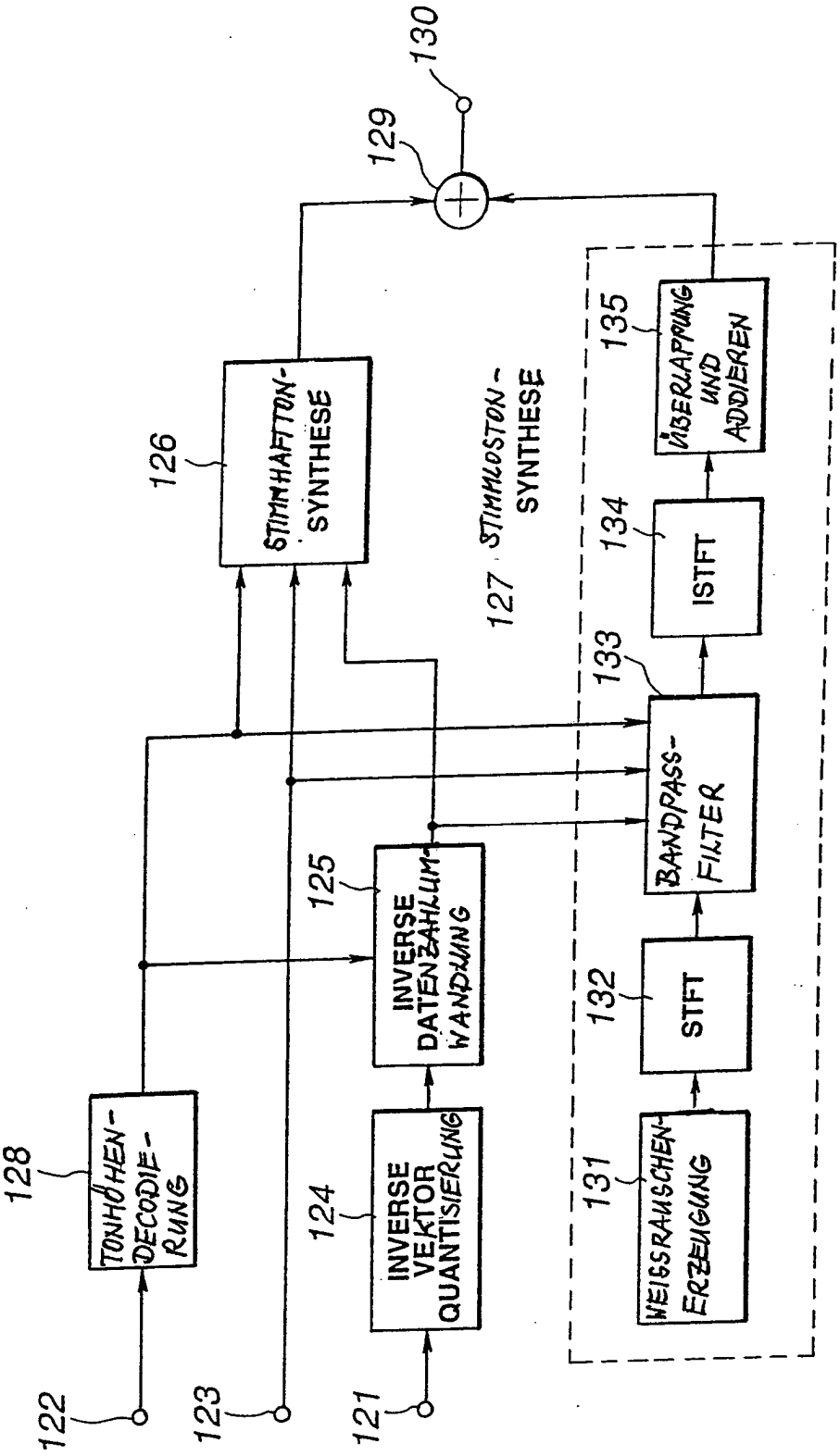

122
123
121
128
TONHÖHEN-DECODIE-RUNG
124
INVERSE VEKTOR QUANTISIERUNG
125
INVERSE DATENZAHLUM-WANDLUNG
126
STIMMHAFTTON-SYNTHESE
127
STIMMLOSTON-SYNTHESE
129
130
131
WEISSRAUSCHEN-ERZEUGUNG
132
STFT
133
BANDPASS-FILTER
134
ISTFT
135
ÜBERLAPPUNG UND ADDIEREN

FIG.6

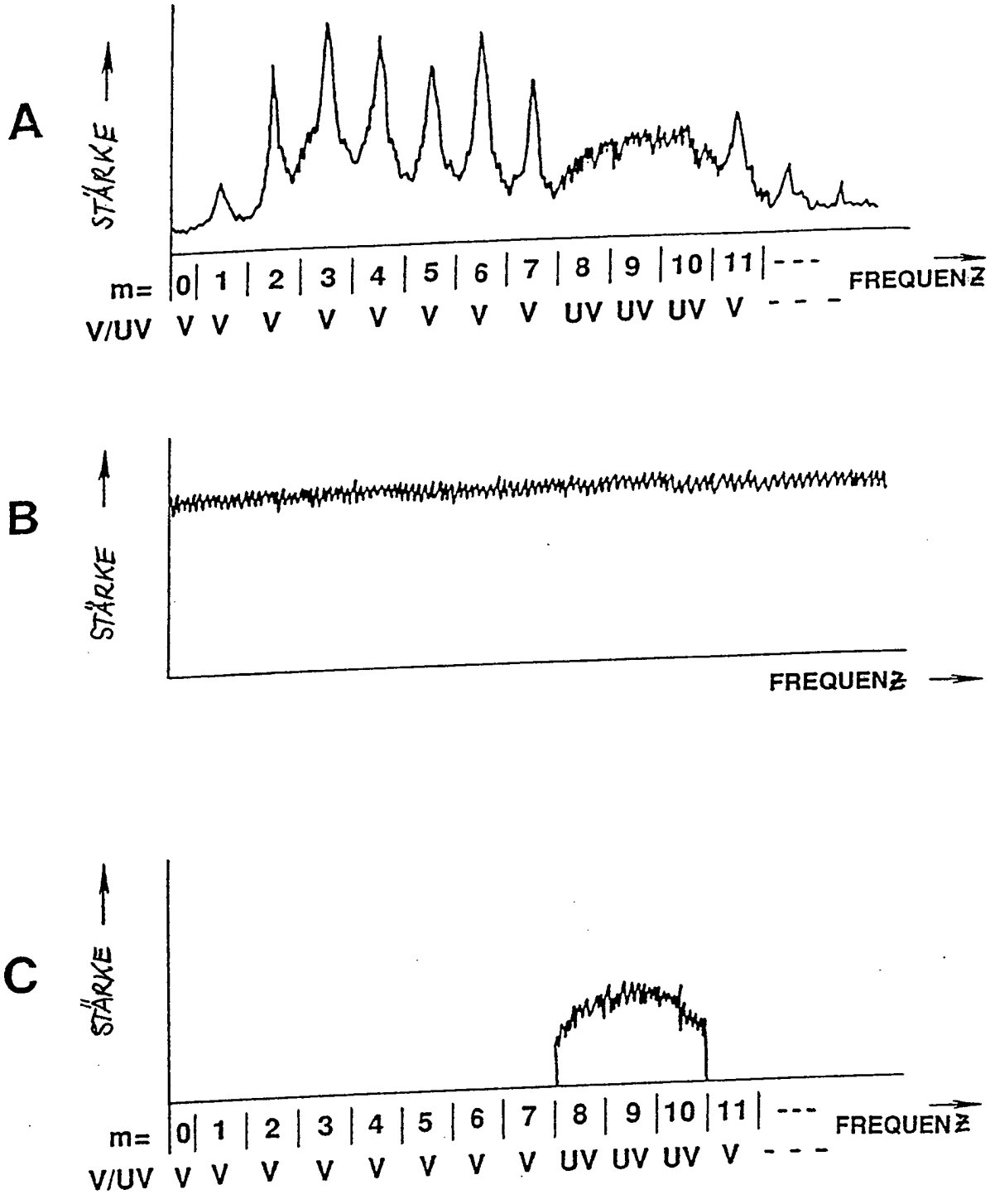

A
STÄRKE
m= 0 1 2 3 4 5 6 7 8 9 10 11 ---
FREQUENZ
V/UV V V V V V V V UV UV UV V - - -
B
STÄRKE
FREQUENZ
C
STÄRKE
m= 0 1 2 3 4 5 6 7 8 9 10 11 ---
FREQUENZ
V/UV V V V V V V V UV UV UV V - - -

FIG.7

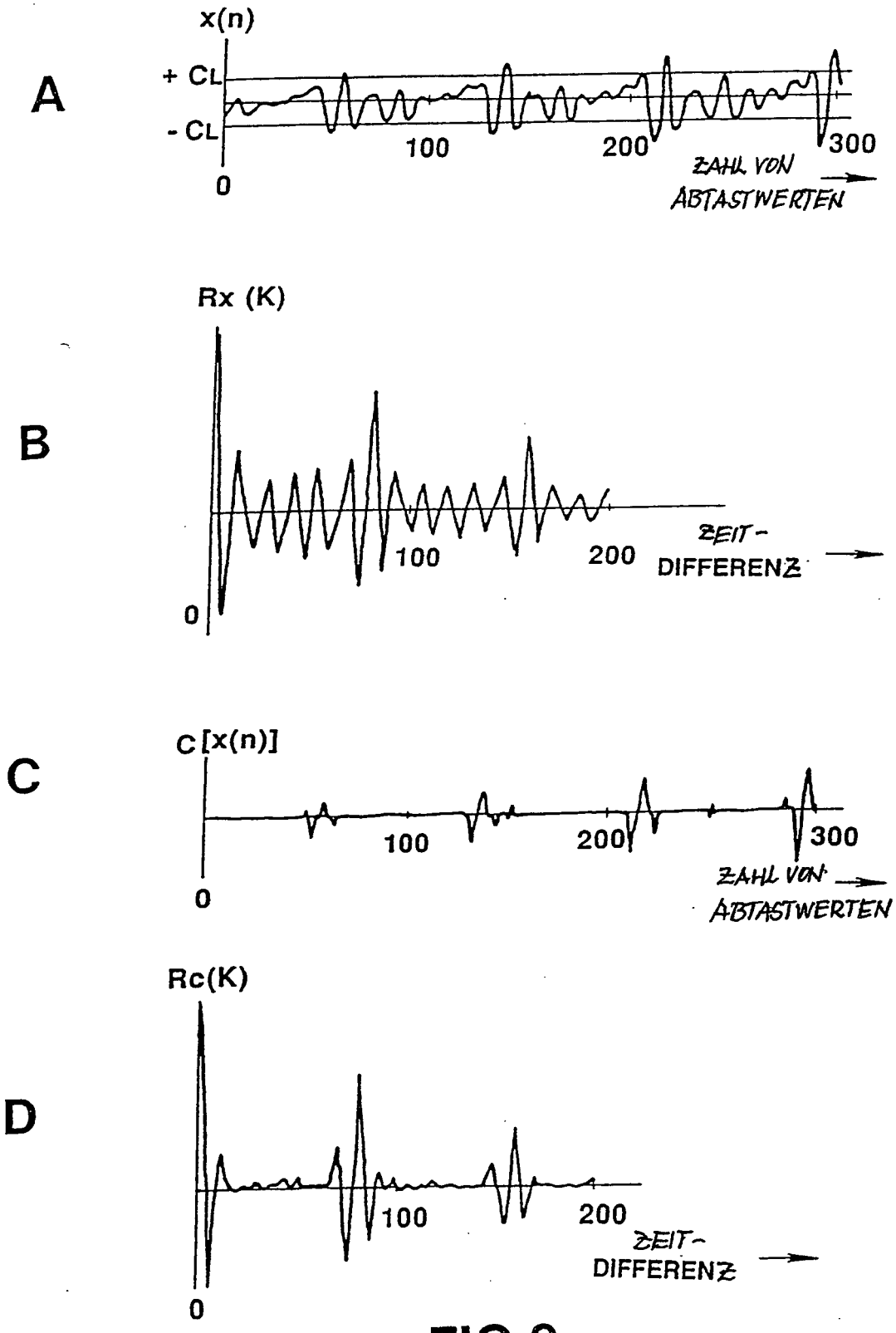

A
x(n)
+ CL
- CL
0
100
200
300
ZAHL VON
ABTASTWERTEN
B
Rx (K)
0
100
200
ZEIT-
DIFFERENZ
C
C[x(n)]
0
100
200
300
ZAHL VON
ABTASTWERTEN
D
Rc(K)
0
100
200
ZEIT-
DIFFERENZ

FIG.8

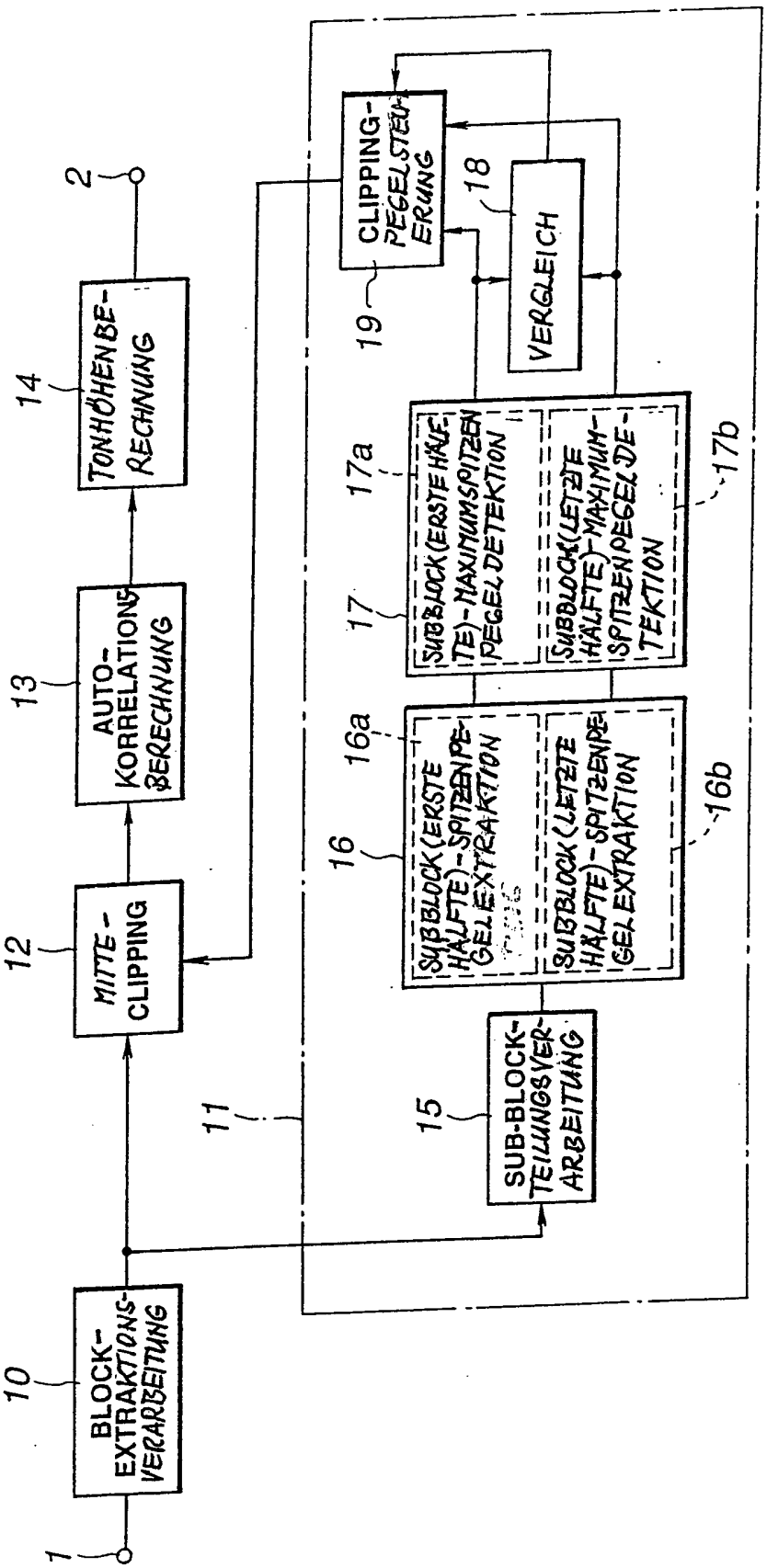

1
2
10
BLOCK-EXTRAKTIONS-VERARBEITUNG
11
12
MITTE-CLIPPING
13
AUTO-KORRELATIONS-BERECHNUNG
14
TONHÖHEN-BE-RECHNUNG
15
SUB-BLOCK-TEILUNGS-VERARBEITUNG
16
16a
SUBBLOCK (ERSTE HÄLFTE) - SPITZENPEGEL EXTRAKTION
16b
SUBBLOCK (LETZTE HÄLFTE) - SPITZENPEGEL EXTRAKTION
17
17a
SUBBLOCK (ERSTE HÄLFTE) - MAXIMUMSPITZEN PEGEL DETEKTION
17b
SUBBLOCK (LETZTE HÄLFTE) - MAXIMUM-SPITZEN PEGEL DETEKTION
18
VERGLEICH
19
CLIPPING-PEGELSTEUERUNG

FIG.9

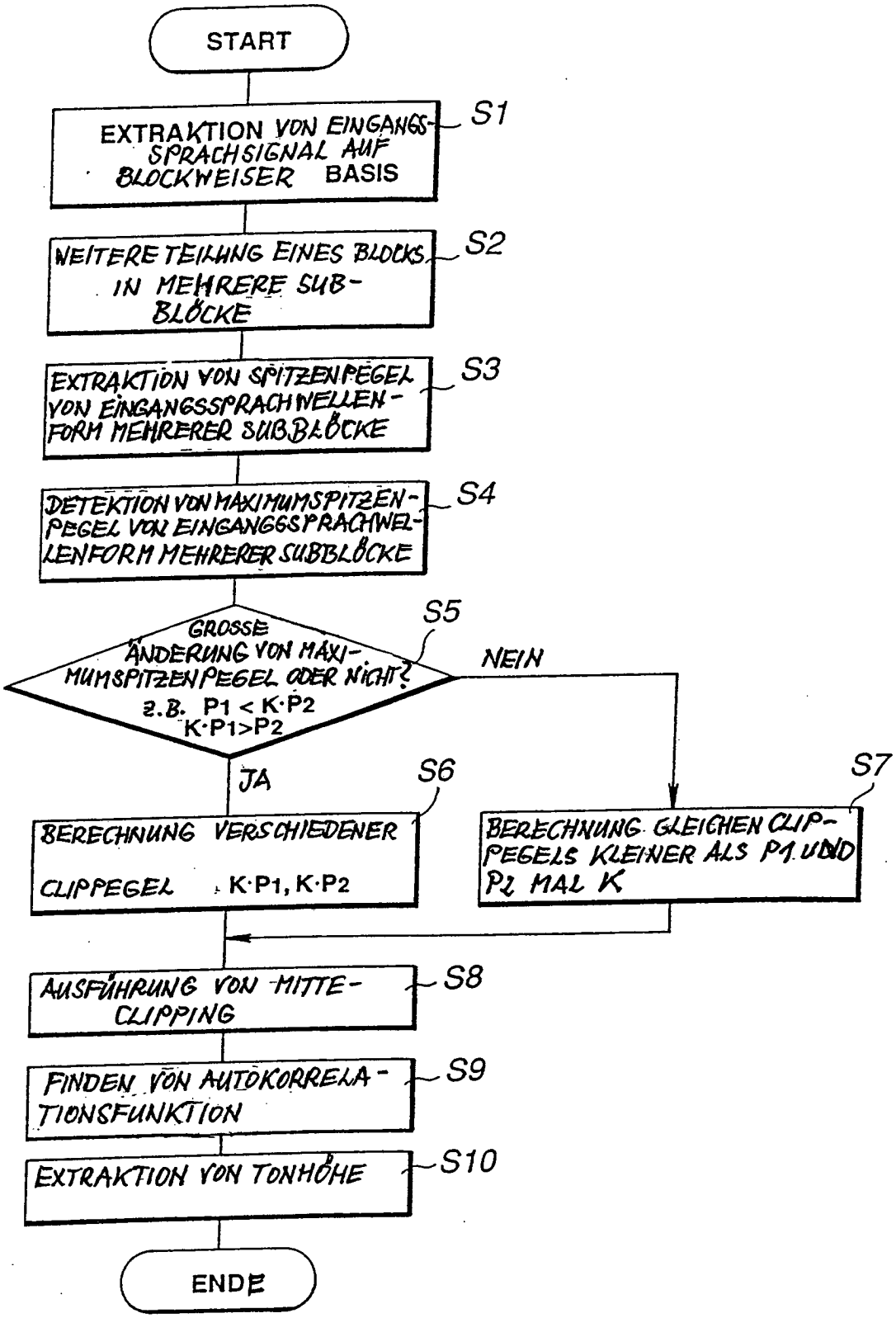

START
EXTRAKTION VON EINGANGS-SPRACHSIGNAL AUF BLOCKWEISER BASIS
S1
WEITERE TEILUNG EINES BLOCKS IN MEHRERE SUB-BLÖCKE
S2
EXTRAKTION VON SPITZENPEGEL VON EINGANGSSPRACHWELLEN-FORM MEHRERER SUBBLÖCKE
S3
DETEKTION VON MAXIMUMSPITZEN-PEGEL VON EINGANGSSPRACHWEL-LENFORM MEHRERER SUBBLÖCKE
S4
GROSSE ÄNDERUNG VON MAXI-MUMSPITZENPEGEL ODER NICHT?
z.B. P1 < K·P2
K·P1>P2
S5
NEIN
JA
S6
BERECHNUNG VERSCHIEDENER CLIPPEGEL K·P1, K·P2
S7
BERECHNUNG GLEICHEN CLIP-PEGELS KLEINER ALS P1 UND P2 MAL K
AUSFÜHRUNG VON MITTE-CLIPPING
S8
FINDEN VON AUTOKORRELA-TIONSFUNKTION
S9
EXTRAKTION VON TONHÖHE
S10
ENDE

FIG.10

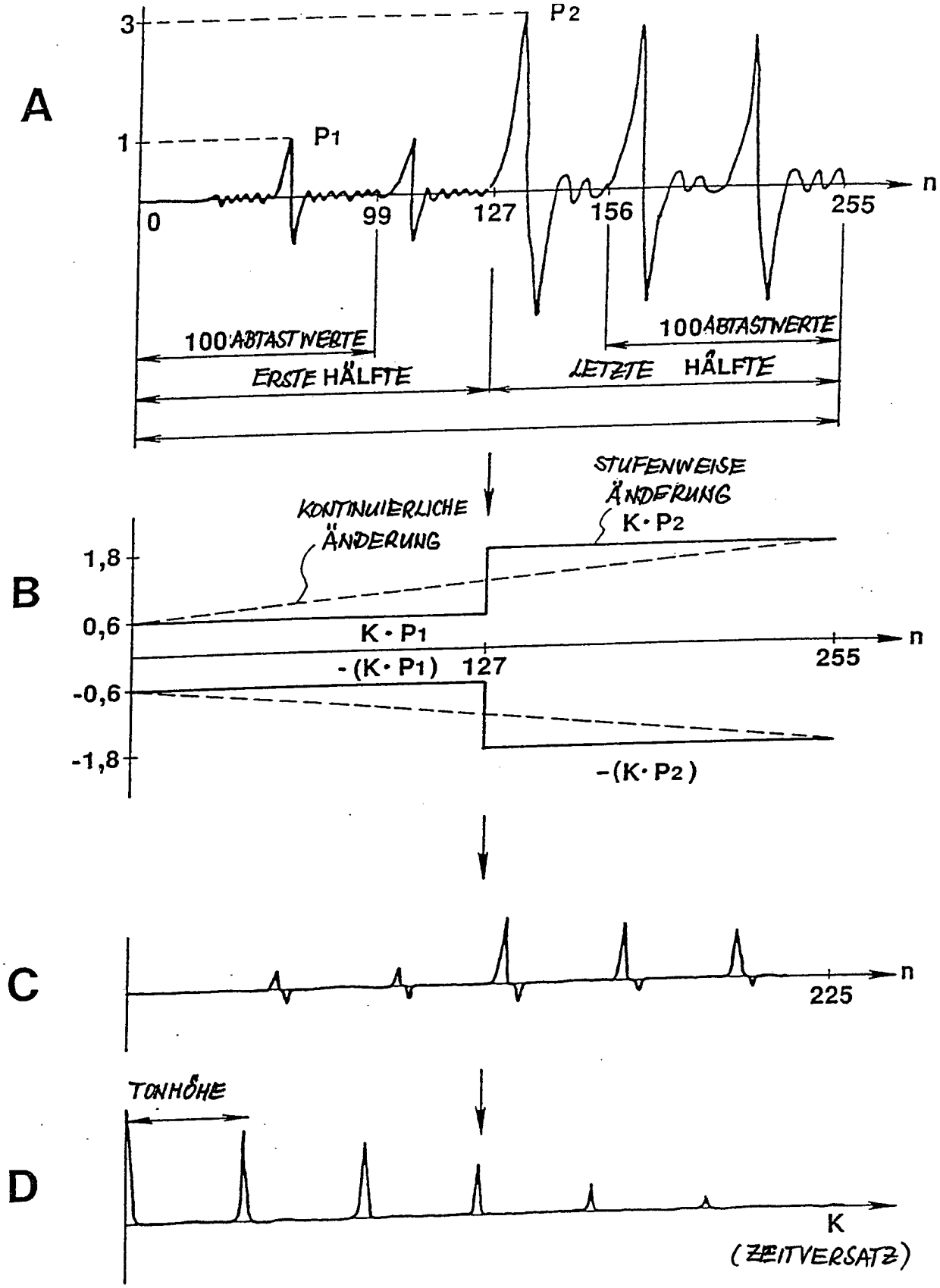

A
3
1
P1
P2
0
99
127
156
255
n
100 ABTASTWERTE
ERSTE HÄLFTE
100 ABTASTWERTE
LETZTE HÄLFTE
B
KONTINUIERLICHE ÄNDERUNG
STUFENWEISE ÄNDERUNG
K · P2
1,8
0,6
K · P1
- (K · P1)
127
255
n
-0,6
-1,8
-(K · P2)
C
225
n
D
TONHÖHE
K
(ZEITVERSATZ)

FIG.11

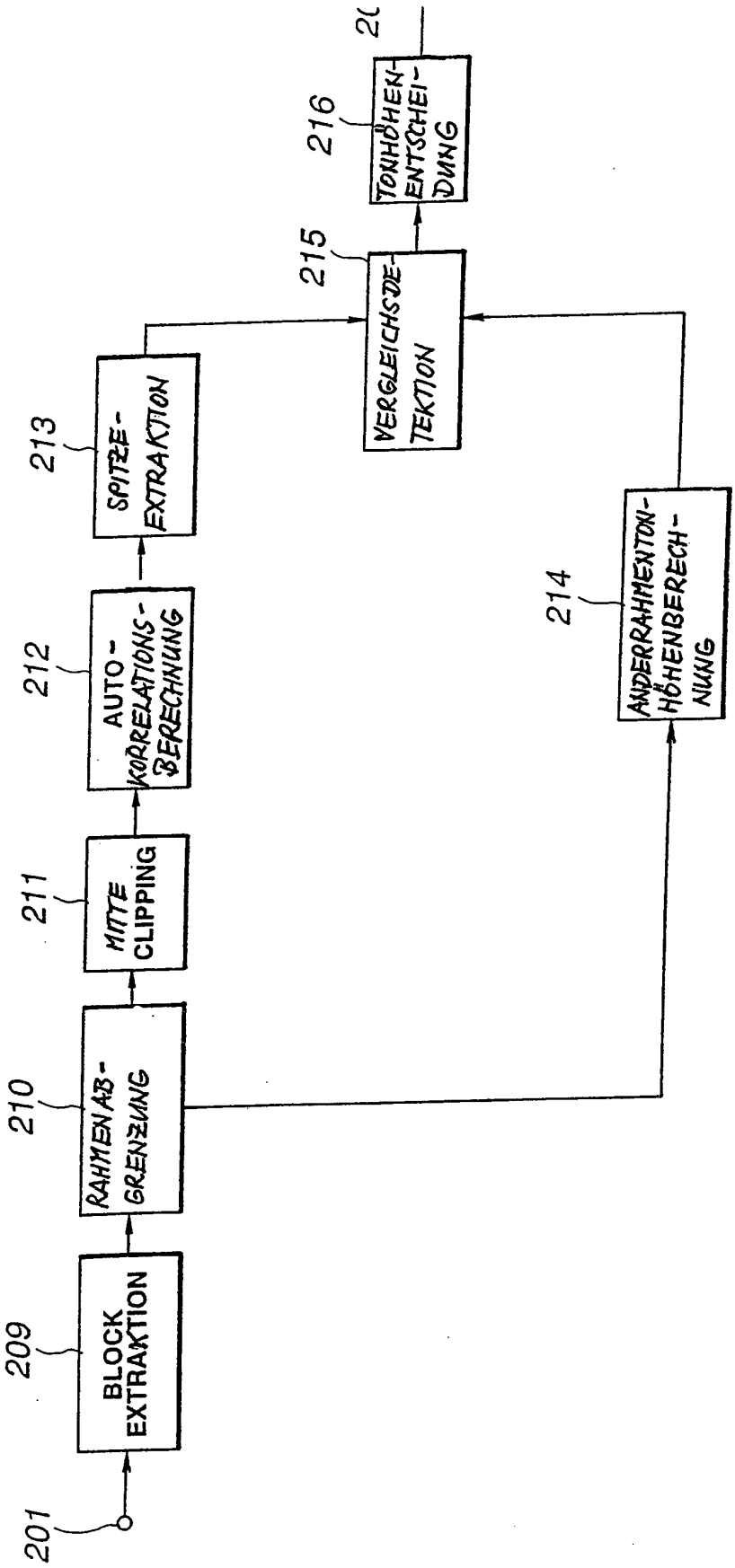

201
BLOCK EXTRAKTION
209
RAHMEN AB-GRENZUNG
210
MITTE CLIPPING
211
AUTO-KORRELATIONS-BERECHNUNG
212
SPITZE-EXTRAKTION
213
ANDERRAHMEN TON-HÖHENBERECH-NUNG
214
VERGLEICHSDE-TEKTION
215
TONHÖHEN-ENTSCHEI-DUNG
216

FIG.12

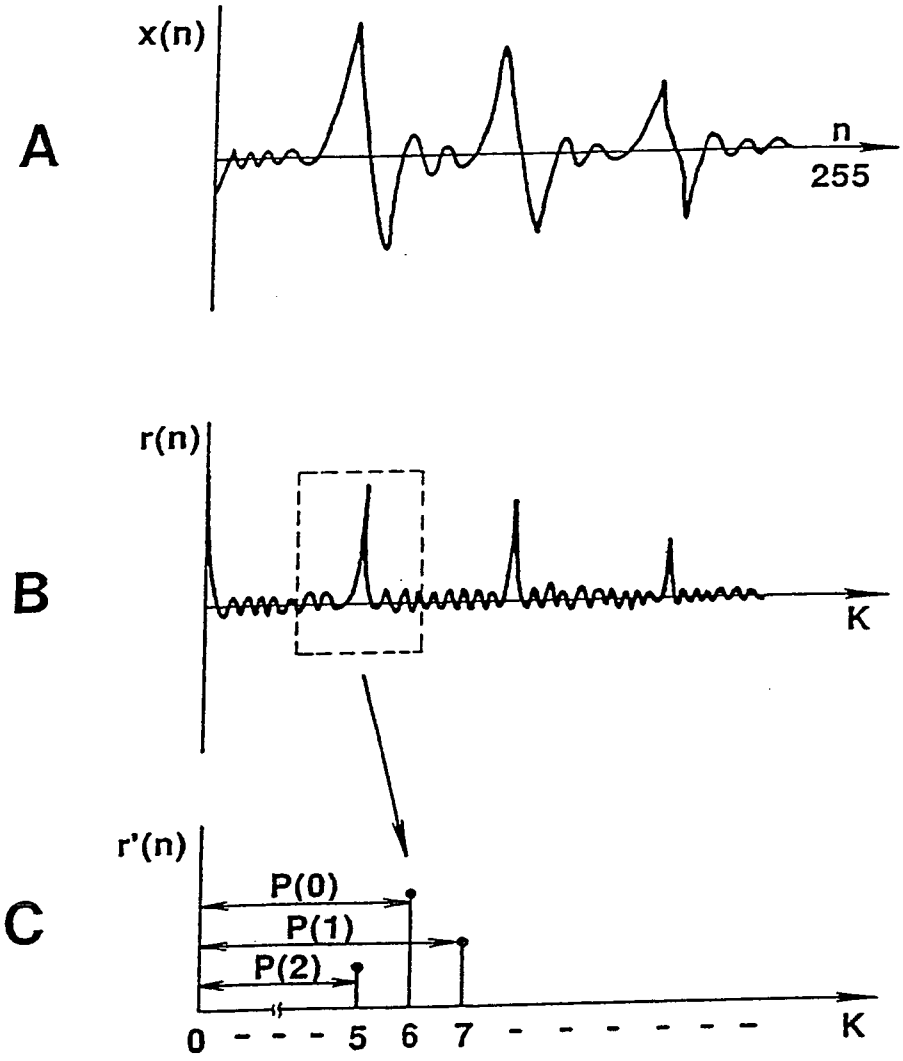


FIG.13

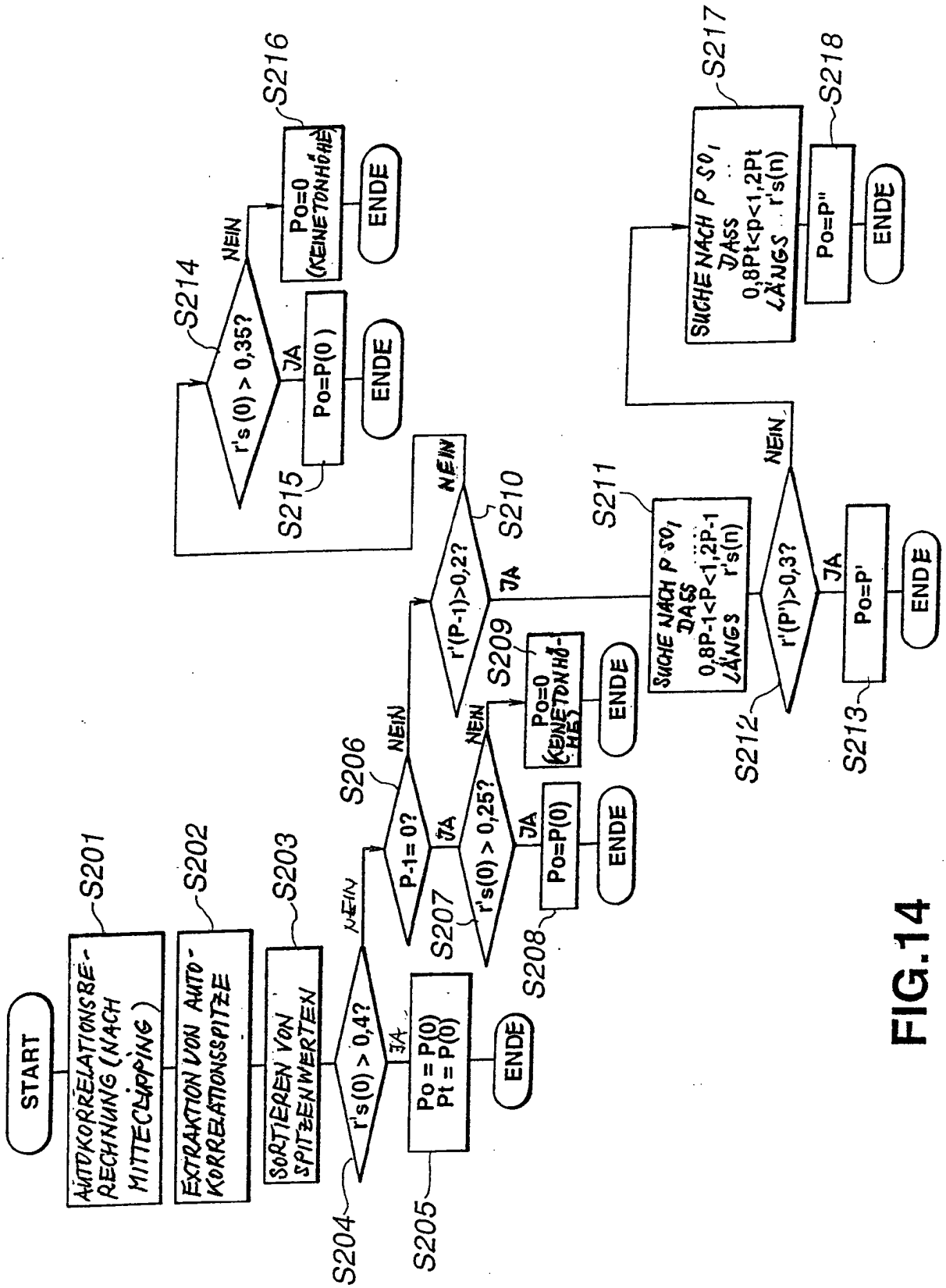

START
S201
AUTOKORRELATIONSBE-RECHNUNG (NACH MITTECLIPPING)
S202
EXTRAKTION VON AUTO-KORRELATIONSSPITZE
S203
SORTIEREN VON SPITZENWERTEN
S204
r's(0) > 0,4?
NEIN
JA
S205
Po = P(0)
Pt = P(0)
ENDE
S206
P-1= 0?
NEIN
JA
S207
r's(0) > 0,25?
NEIN
JA
S208
Po=P(0)
ENDE
S209
Po=0 (KEINE TONHÖHE)
ENDE
S210
r'(P-1)>0,2?
NEIN
JA
S211
SUCHE NACH P SO, DASS 0,8P-1<P<1,2P-1 LÄNGS r's(n)
S212
r'(P')>0,3?
NEIN
JA
S213
Po=P'
ENDE
S214
r's (0) > 0,35?
NEIN
JA
S215
Po=P(0)
ENDE
S216
Po=0 (KEINE TONHÖHE)
ENDE
S217
SUCHE NACH P SO, DASS 0,8Pt<p<1,2Pt LÄNGS r's(n)
S218
Po=P''
ENDE

FIG.14

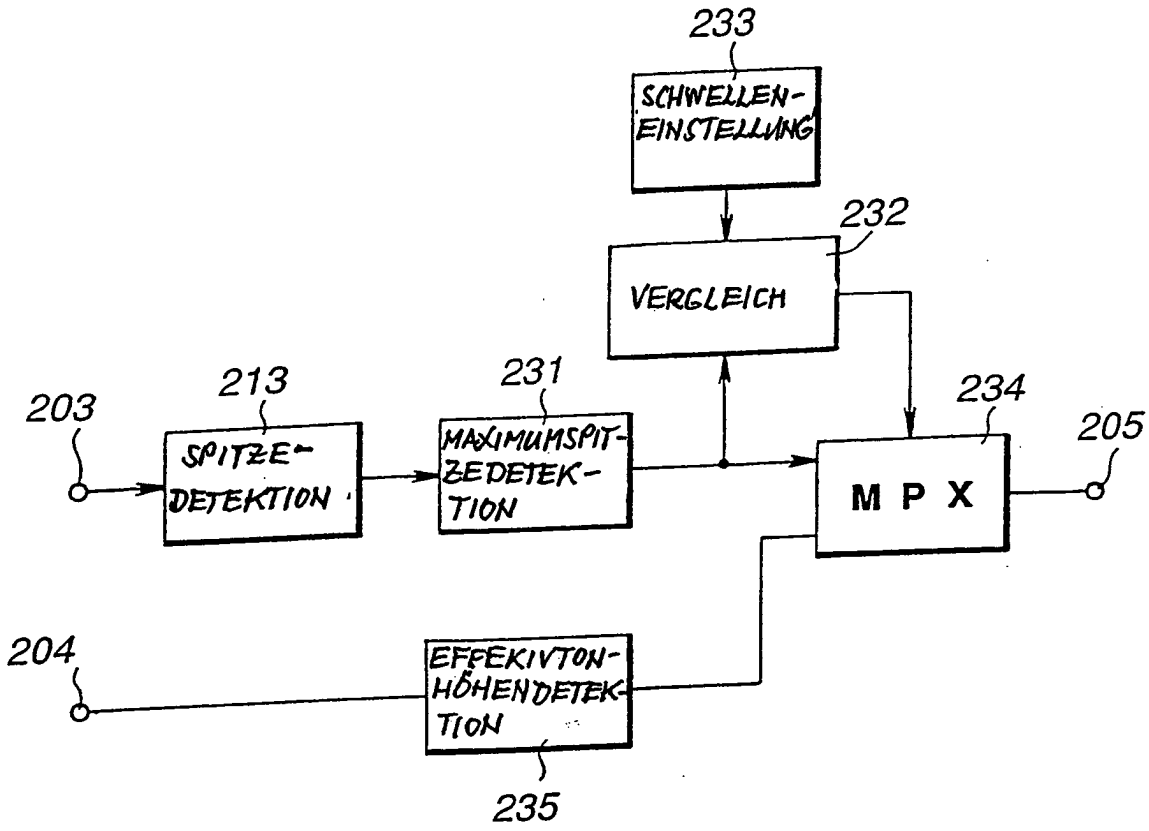

233
SCHWELLEN-EINSTELLUNG
232
VERGLEICH
213
231
234
203
205
SPITZE-DETEKTION
MAXIMUMSPIT-ZEDETEK-TION
M P X
204
EFFEKIVTON-HÖHENDETEK-TION
235

FIG.15

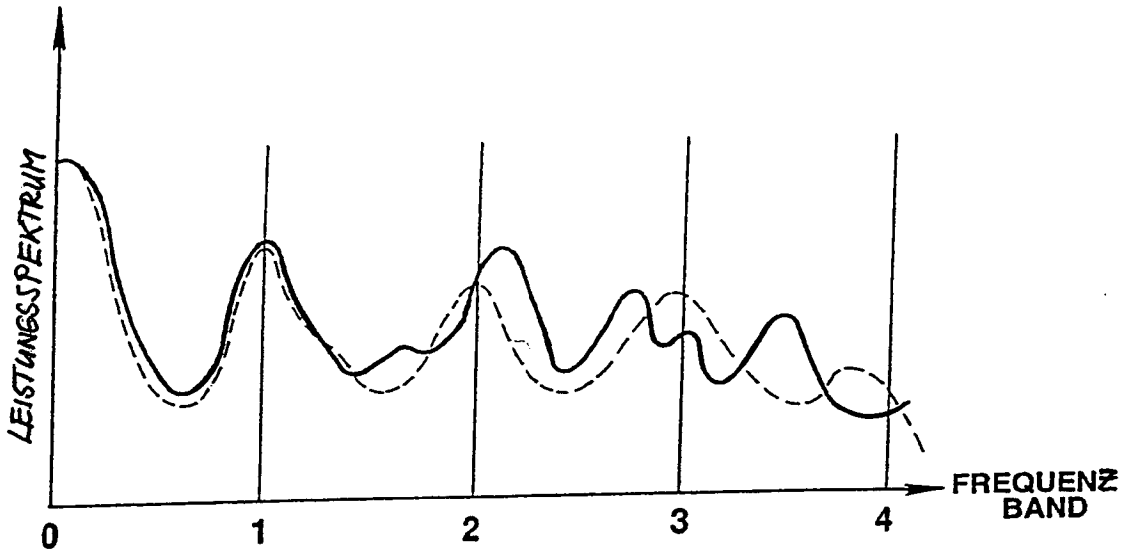


FIG.16

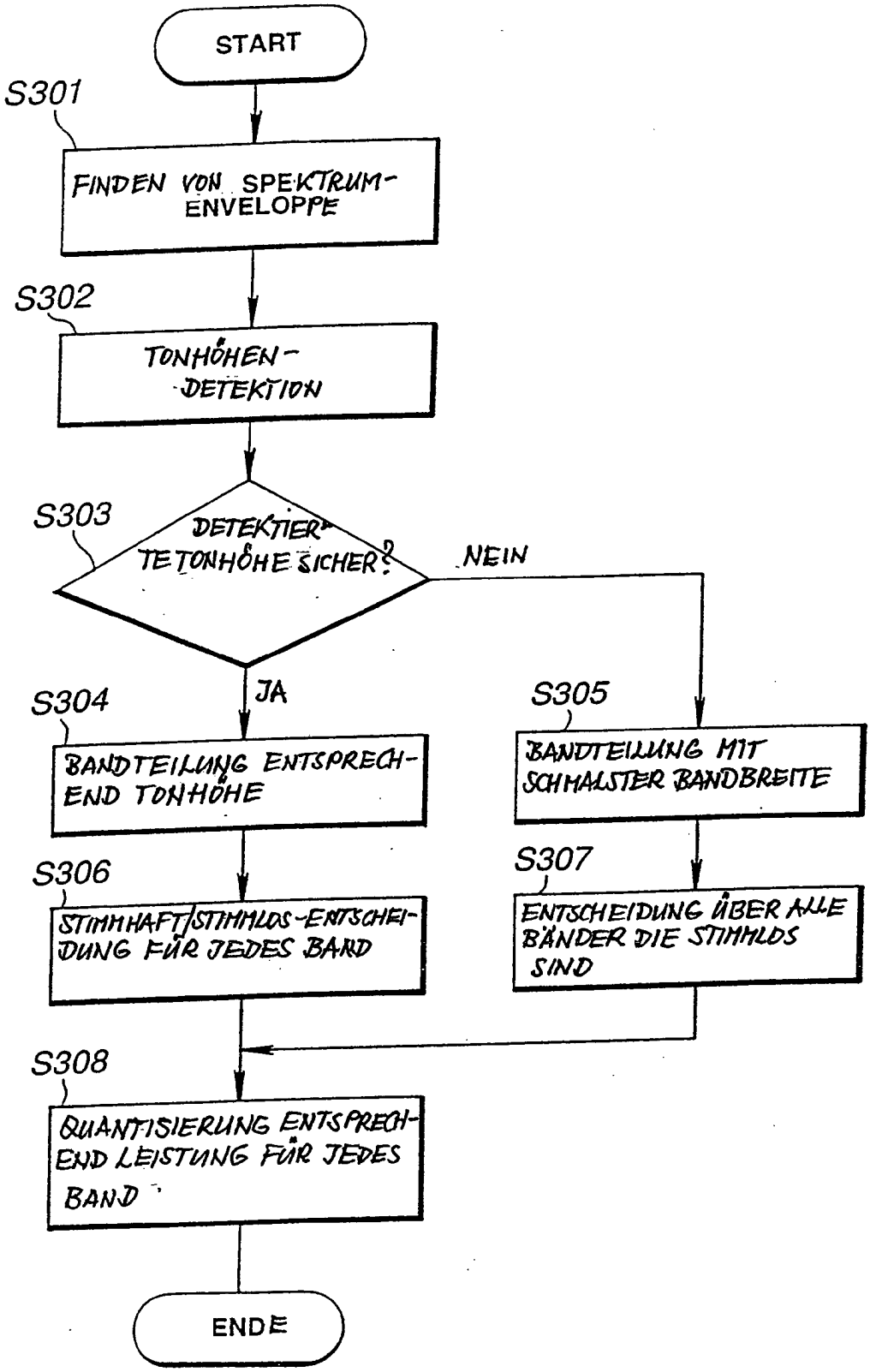

START
S301
FINDEN VON SPEKTRUM-ENVELOPPE
S302
TONHÖHEN-DETEKTION
S303
DETEKTIERTE TONHÖHE SICHER?
NEIN
JA
S304
BANDTEILUNG ENTSPRECHEND TONHÖHE
S305
BANDTEILUNG MIT SCHMALSTER BANDBREITE
S306
STIMMHAFT/STIMMLOS-ENTSCHEIDUNG FÜR JEDES BAND
S307
ENTSCHEIDUNG ÜBER ALLE BÄNDER DIE STIMMLOS SIND
S308
QUANTISIERUNG ENTSPRECHEND LEISTUNG FÜR JEDES BAND
ENDE

FIG.17

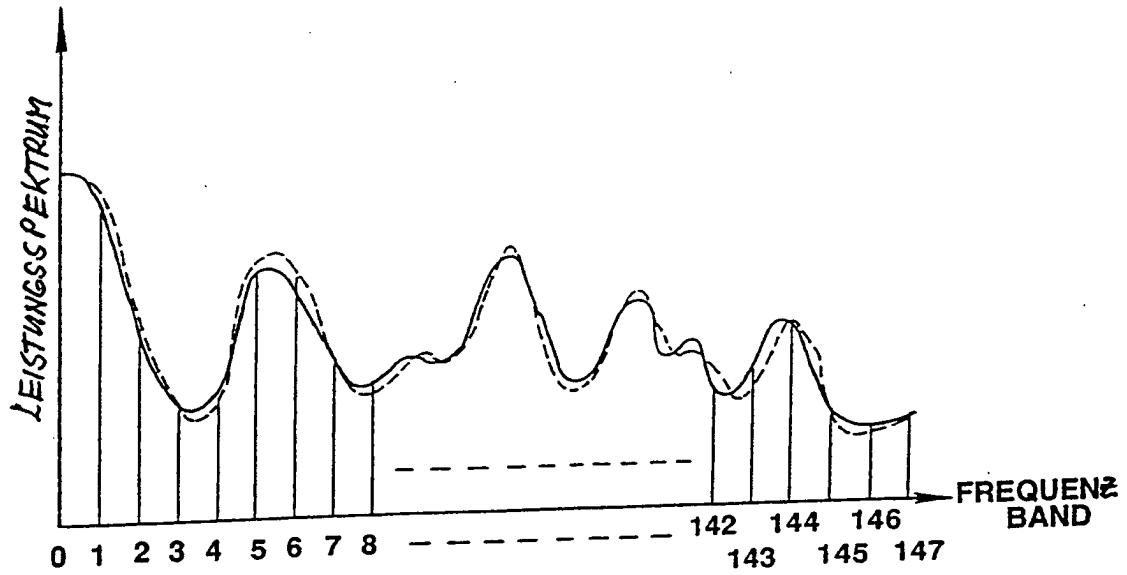

LEISTUNGSSPEKTRUM
FREQUENZ BAND
0 1 2 3 4 5 6 7 8
142 144 146
143 145 147

FIG.18

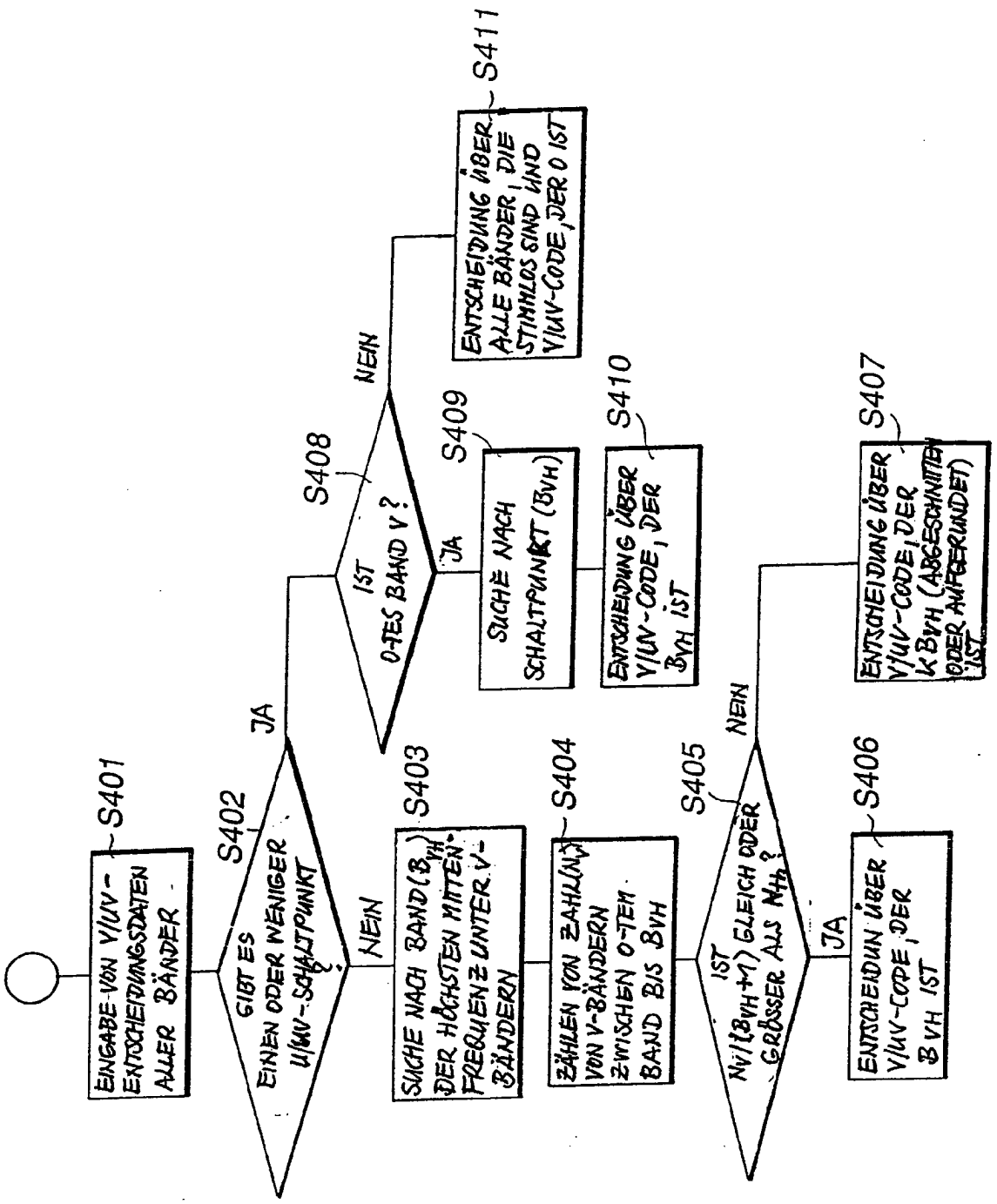

S401
EINGABE VON V/UV-ENTSCHEIDUNGSDATEN ALLER BÄNDER
S402
GIBT ES EINEN ODER MEHRERE U/MV-SCHALTPUNKT?
JA
NEIN
S403
SUCHE NACH BAND (B_{VH}) DER HÖCHSTEN MITTEN-FREQUENZ UNTER V-BÄNDERN
S404
ZÄHLEN VON ZAHL (N_V) VON V-BÄNDERN ZWISCHEN 0-TEM BAND BIS B_{VH}
S405
IST N_V ($\geq B_{VH}+1$) GLEICH ODER GRÖSSER ALS N_{th}?
JA
NEIN
S406
ENTSCHEIDUN ÜBER V/UV-CODE, DER B_{VH} IST
S407
ENTSCHEIDUNG ÜBER V/UV-CODE, DER K_{BVH} (ABGESCHNITTEN ODER AUFGERUNDET) IST
S408
IST 0-TES BAND V?
JA
NEIN
S409
SUCHE NACH SCHALTPUNKT (B_{VH})
S410
ENTSCHEIDUNG ÜBER V/UV-CODE, DER B_{VH} IST
S411
ENTSCHEIDUNG ÜBER ALLE BÄNDER, DIE STIMMLOS SIND UND V/UV-CODE, DER 0 IST

FIG.19

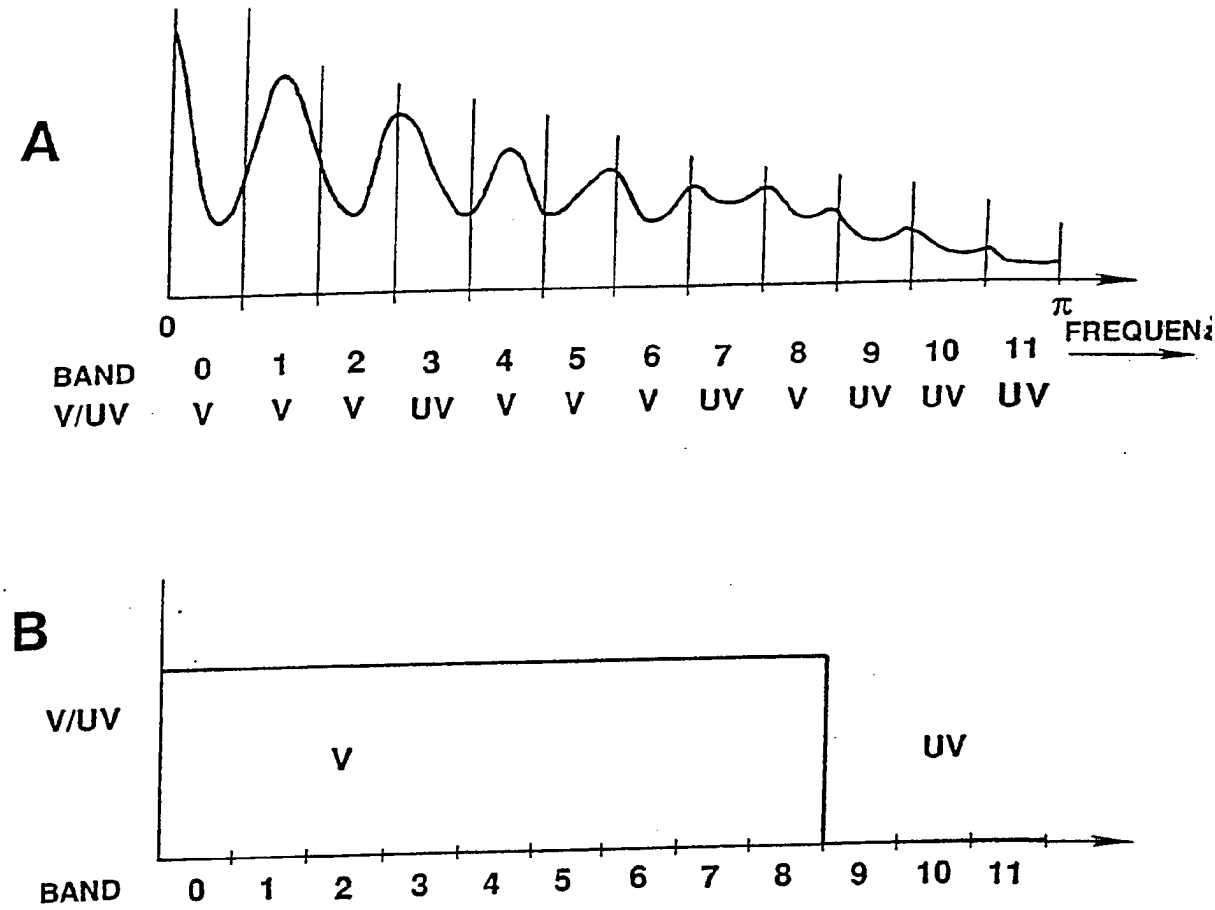

A
0
π
FREQUENZ
BAND 0 1 2 3 4 5 6 7 8 9 10 11
V/UV V V V UV V V V UV V UV UV UV
B
V/UV
V
UV
BAND 0 1 2 3 4 5 6 7 8 9 10 11

FIG.20

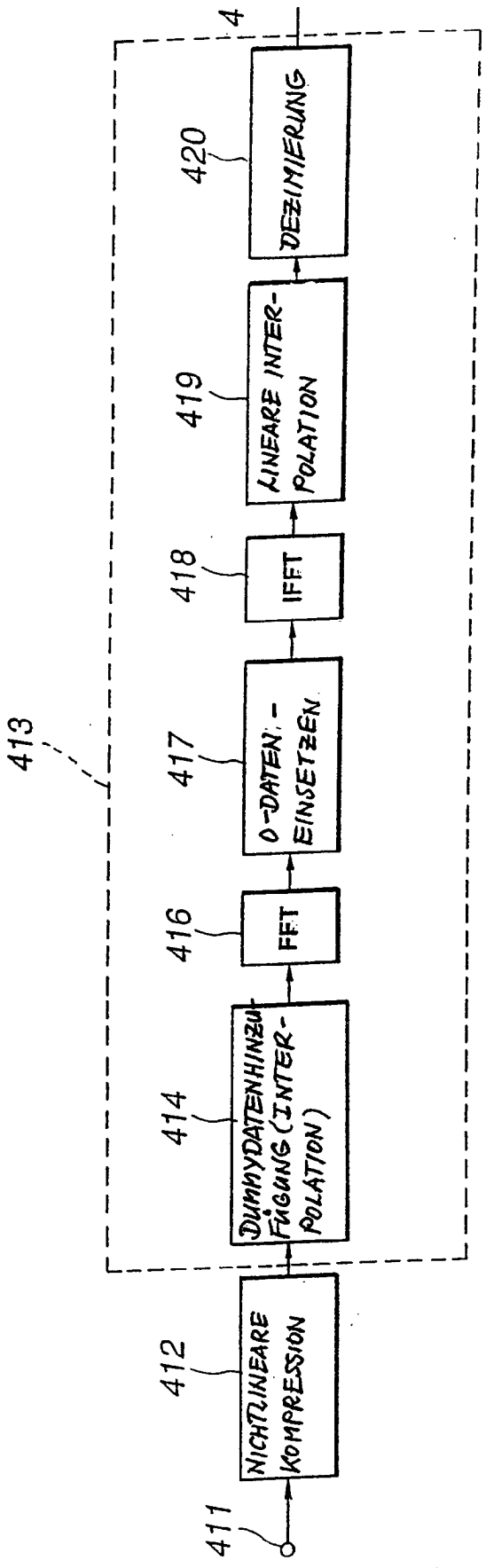

411
412
NICHTLINEARE KOMPRESSION
414
DUMMYDATENHINZUFÜGUNG (INTERPOLATION)
416
FFT
417
0-DATEN-EINSETZEN
418
IFFT
419
LINEARE INTERPOLATION
420
DEZIMIERUNG
413
4

FIG.21

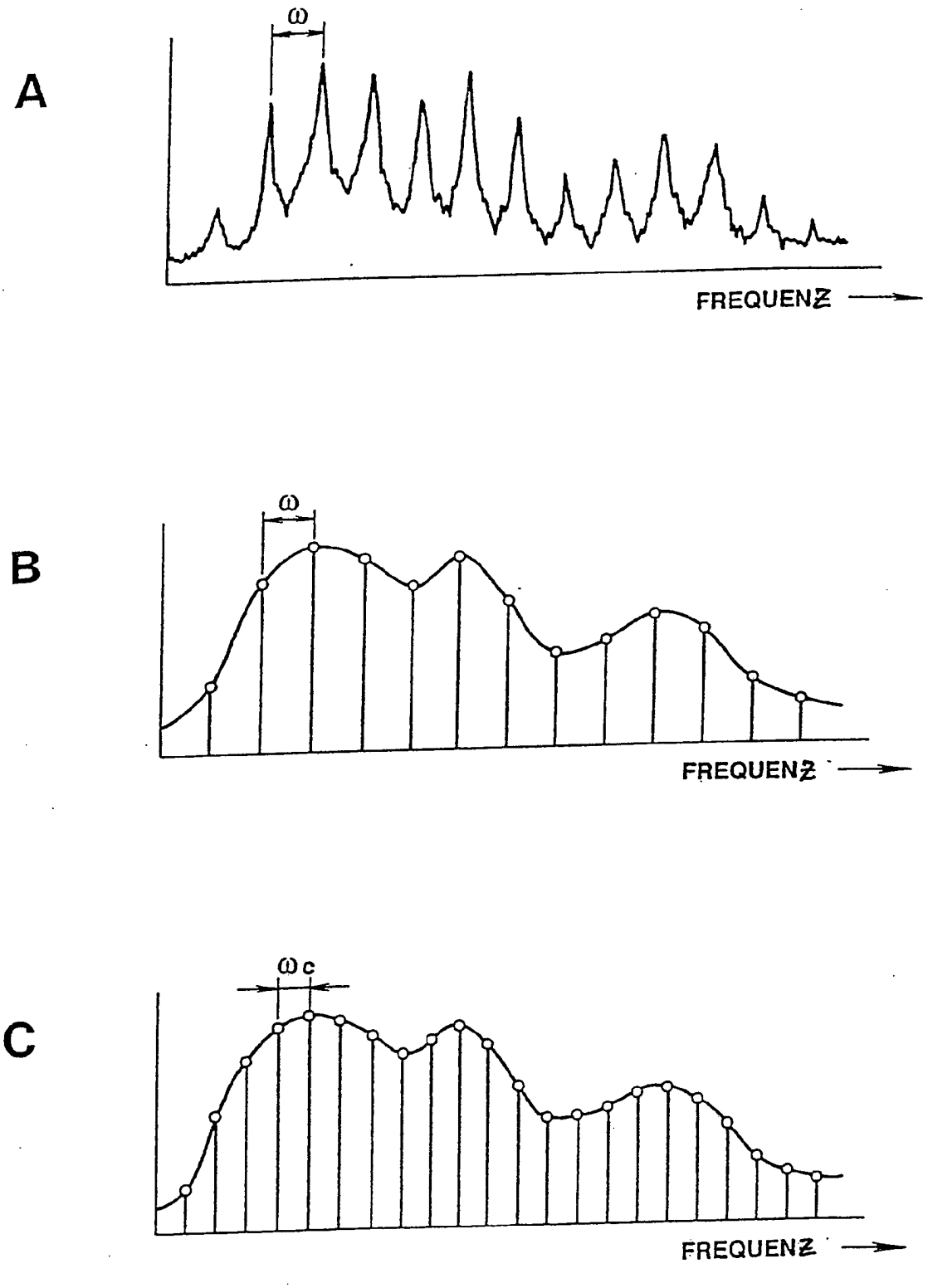


FIG.22

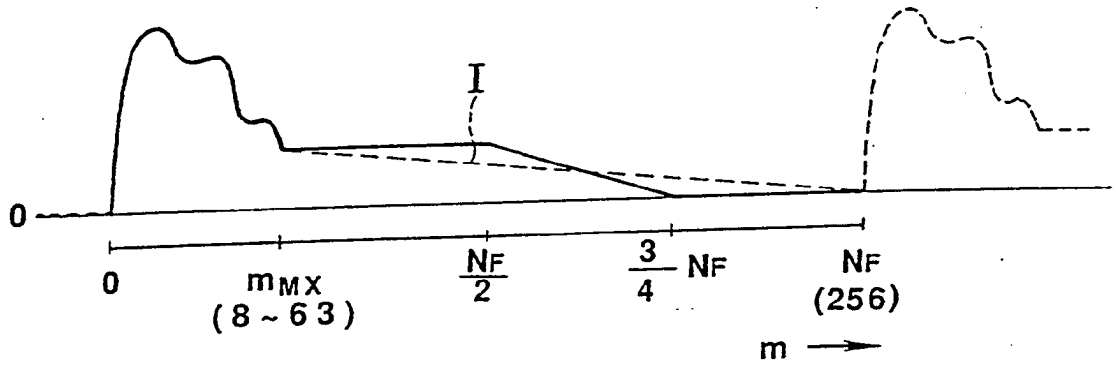


FIG.23

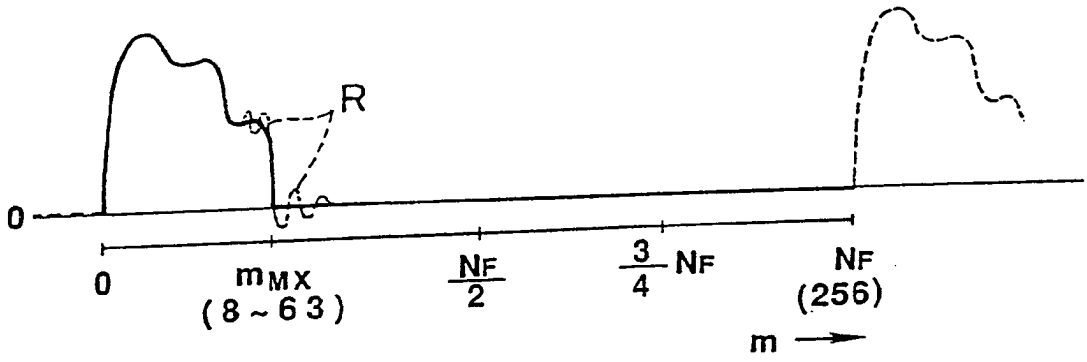


FIG.24

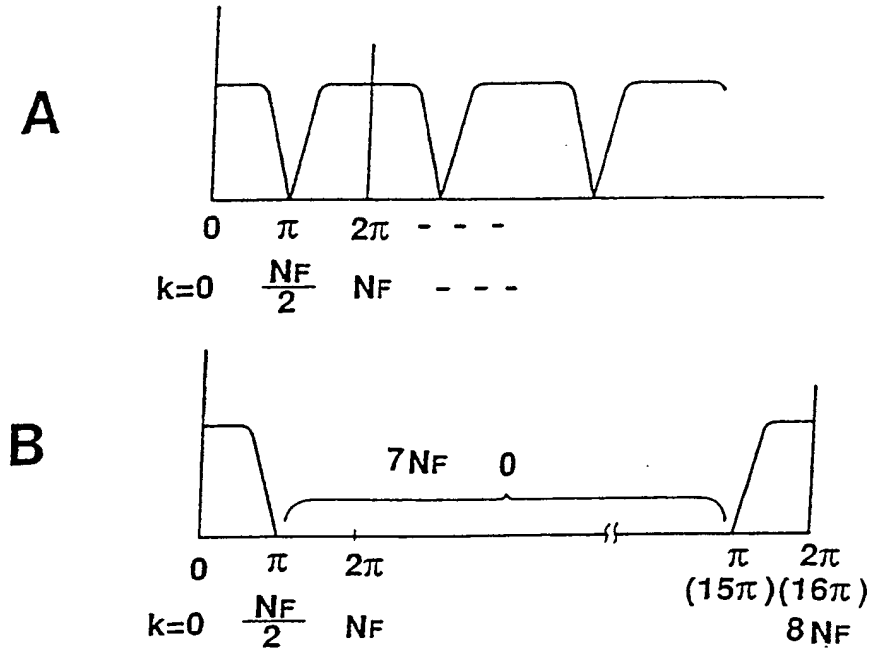

A
0
π
2π
- - -
k=0
NF/2
NF
- - -
B
7NF
0
0
π
2π
π
2π
(15π)(16π)
8NF
k=0
NF/2
NF

FIG.25

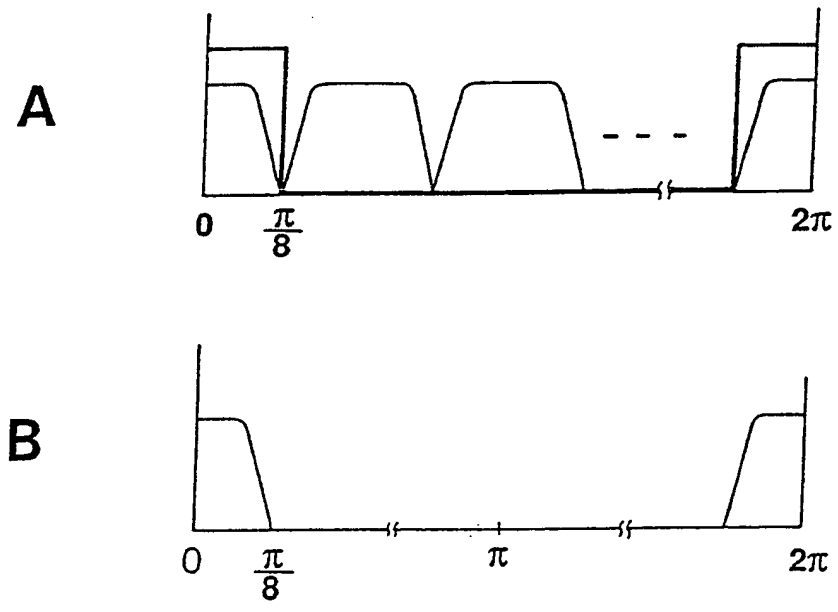

A
0
π/8
- - -
2π
B
0
π/8
π
2π

FIG.26

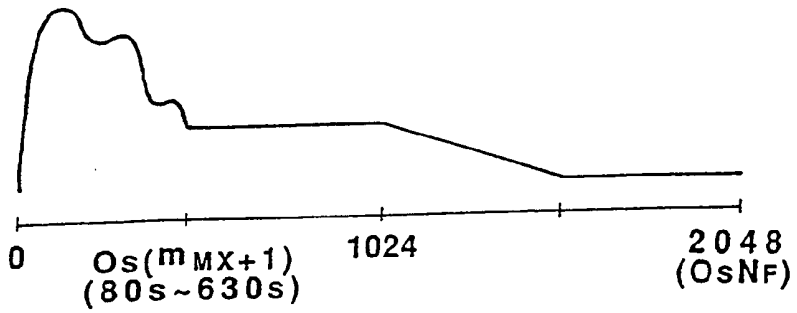


FIG.27

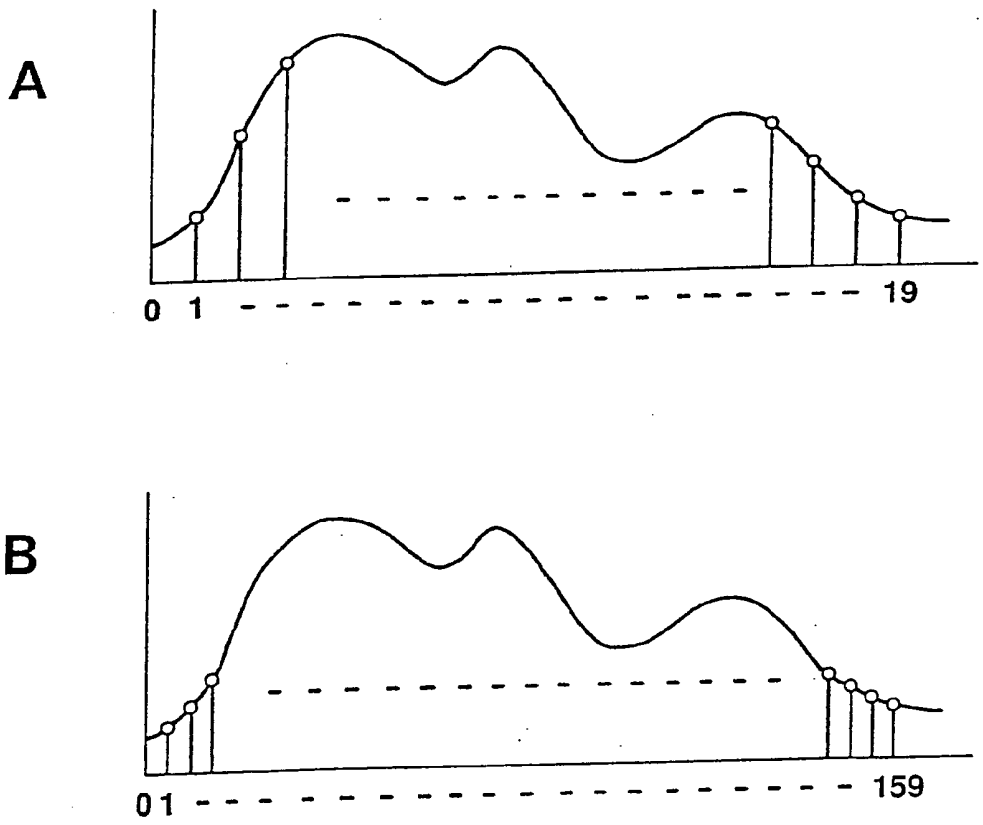


FIG.28

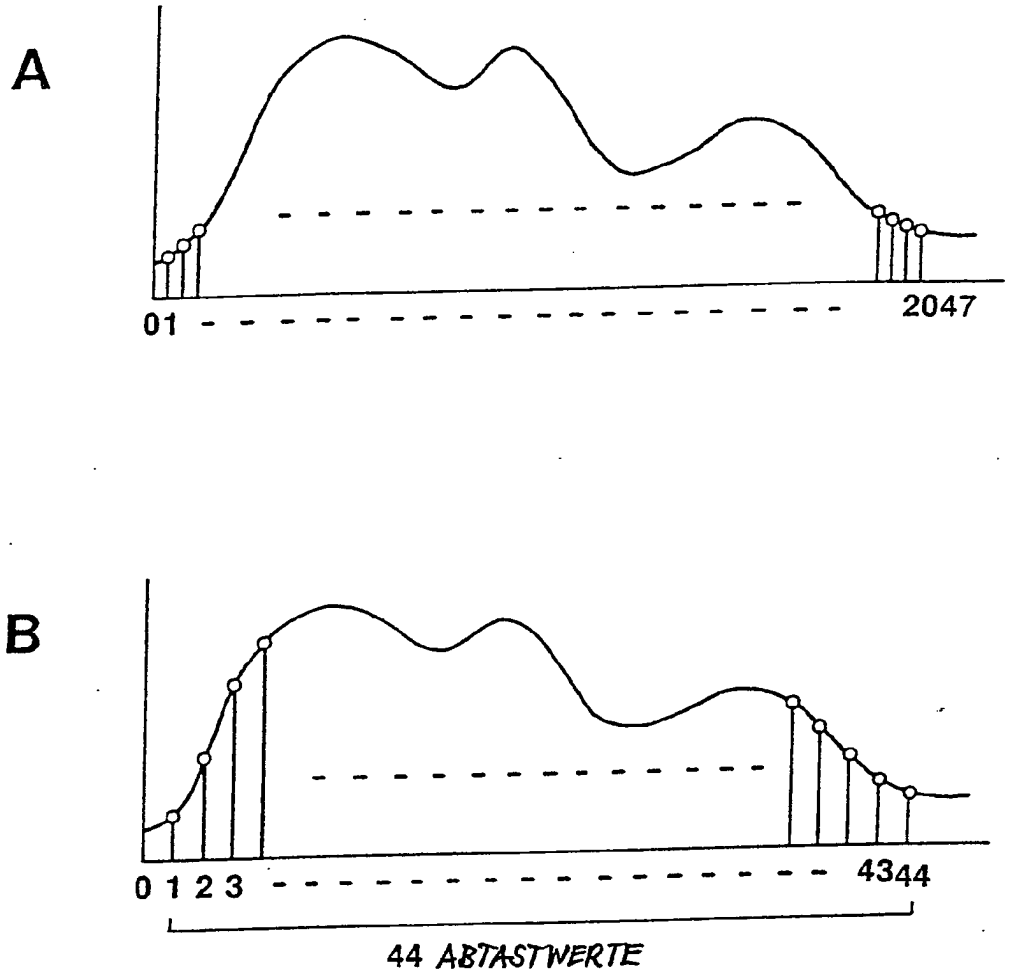


FIG.29

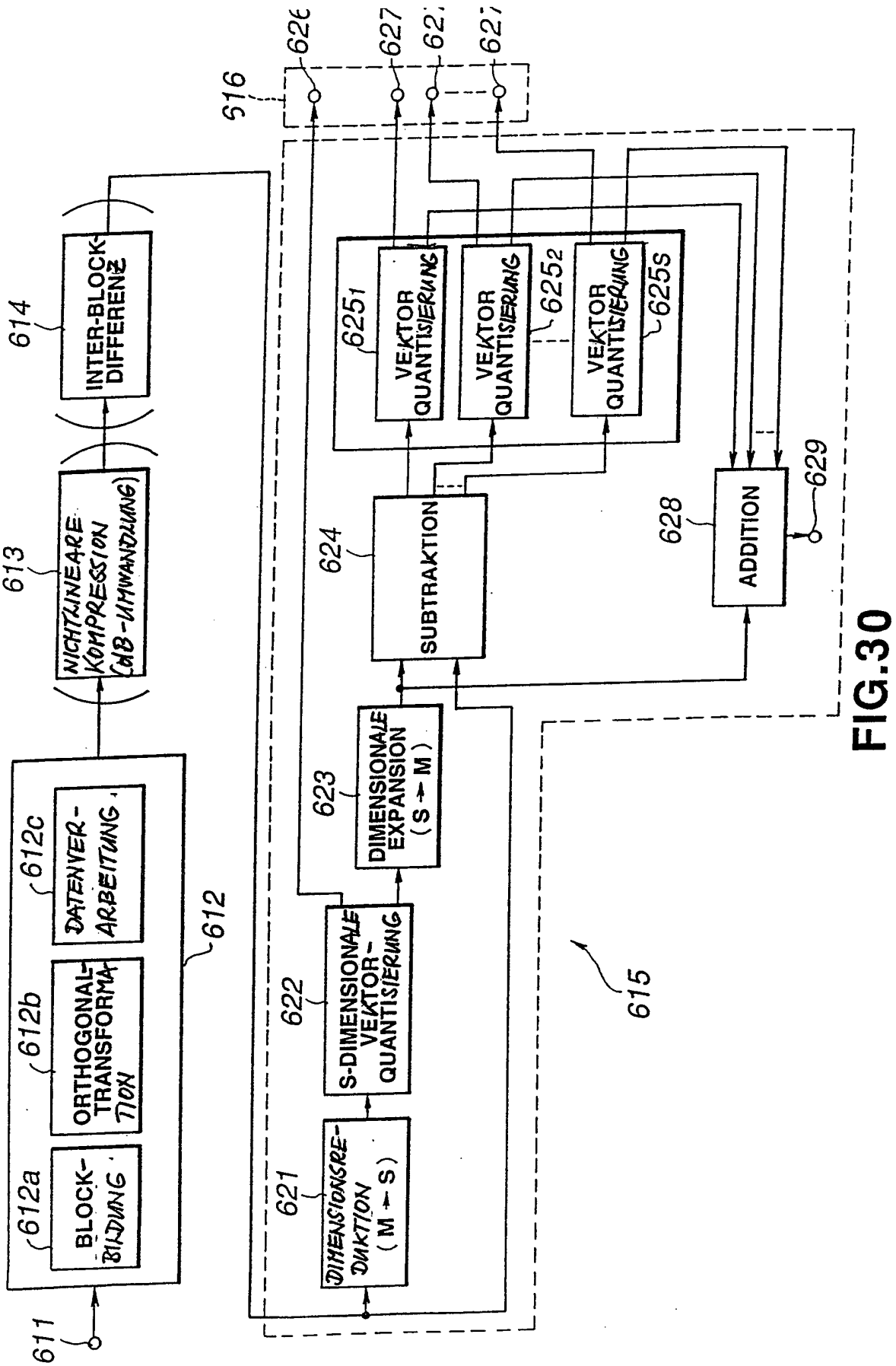

611
612
612a
BLOCK-BILDUNG
612b
ORTHOGONAL-TRANSFORMATION
612c
DATENVER-ARBEITUNG
613
NICHTLINEARE KOMPRESSION (AB-UMWANDLUNG)
614
INTER-BLOCK DIFFERENZ
615
616
621
DIMENSIONSREDUKTION (M → S)
622
S-DIMENSIONALE VEKTOR-QUANTISIERUNG
623
DIMENSIONALE EXPANSION (S → M)
624
SUBTRAKTION
625₁
VEKTOR QUANTISIERUNG
625₂
VEKTOR QUANTISIERUNG
625S
VEKTOR QUANTISIERUNG
626
627
628
ADDITION
629

FIG.30

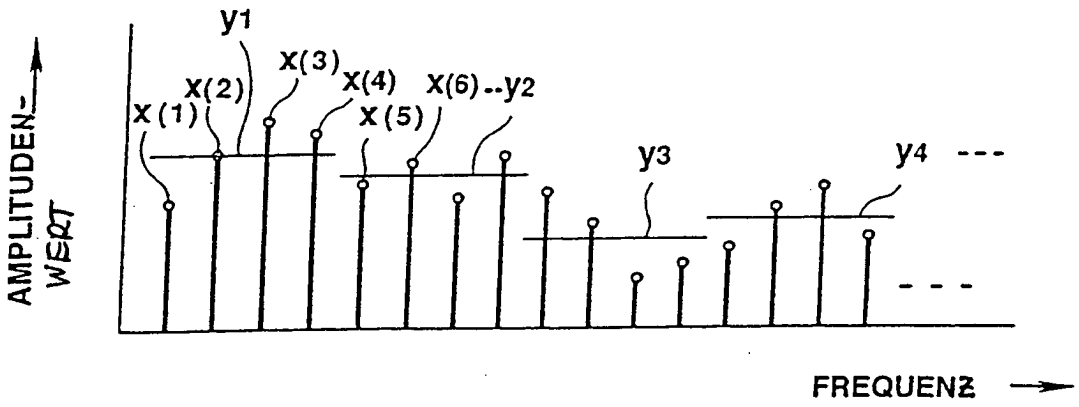


FIG.31

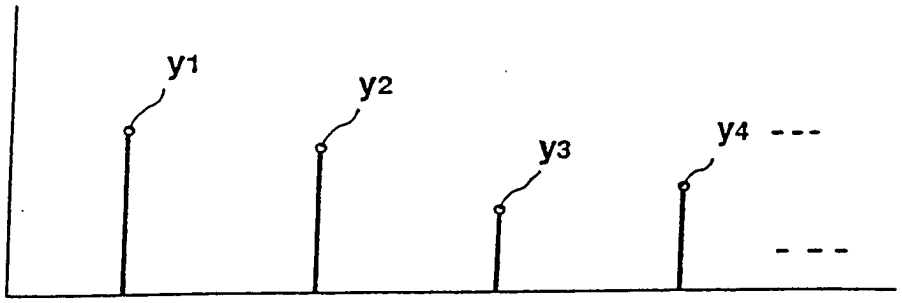


FIG.32

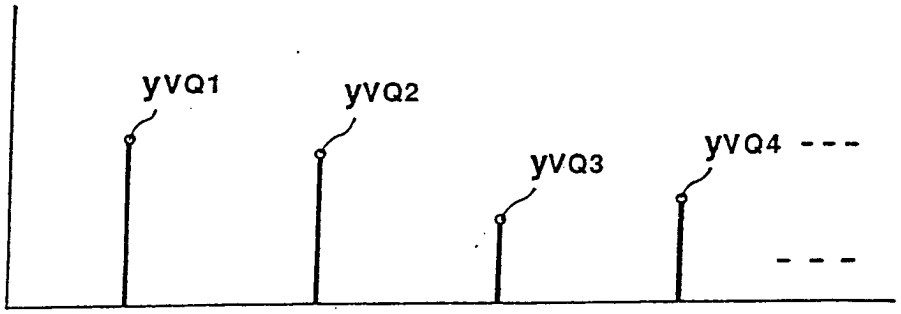


FIG.33

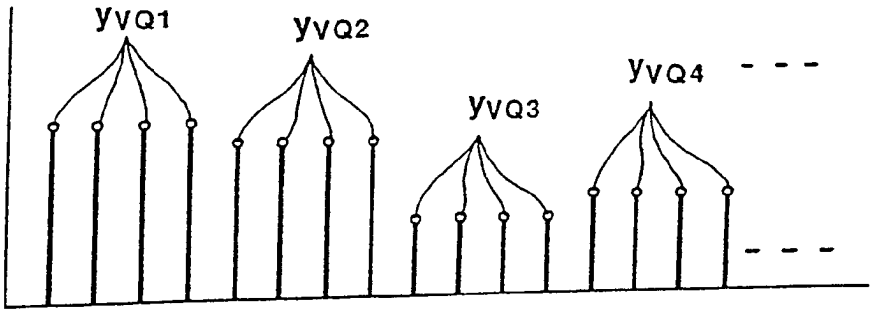


FIG.34

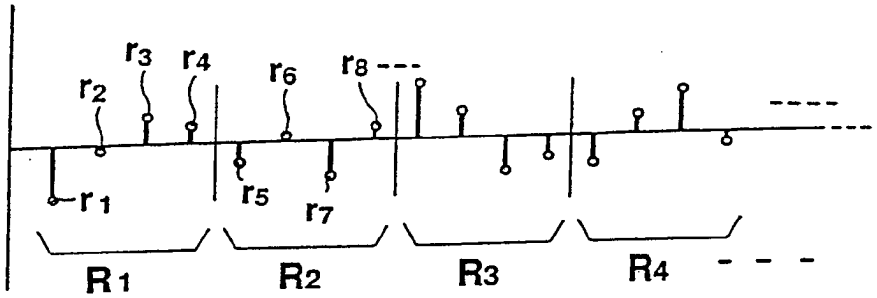


FIG.35

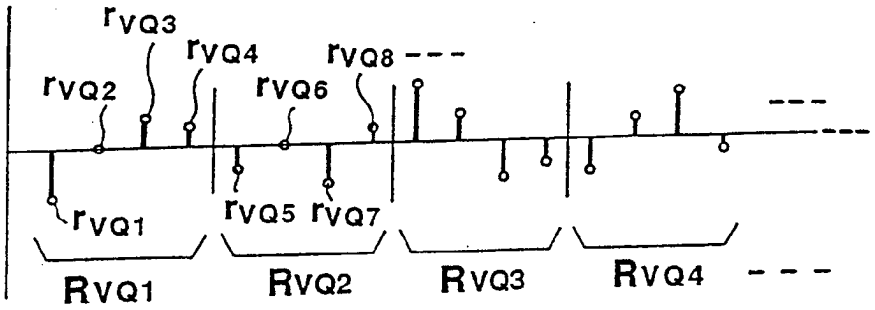


FIG.36

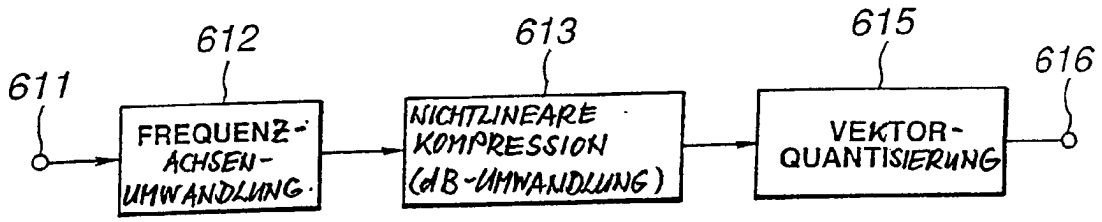

611
612
FREQUENZ-ACHSEN-UMWANDLUNG
613
NICHTLINEARE KOMPRESSION (dB-UMWANDLUNG)
615
VEKTOR-QUANTISIERUNG
616

FIG.37

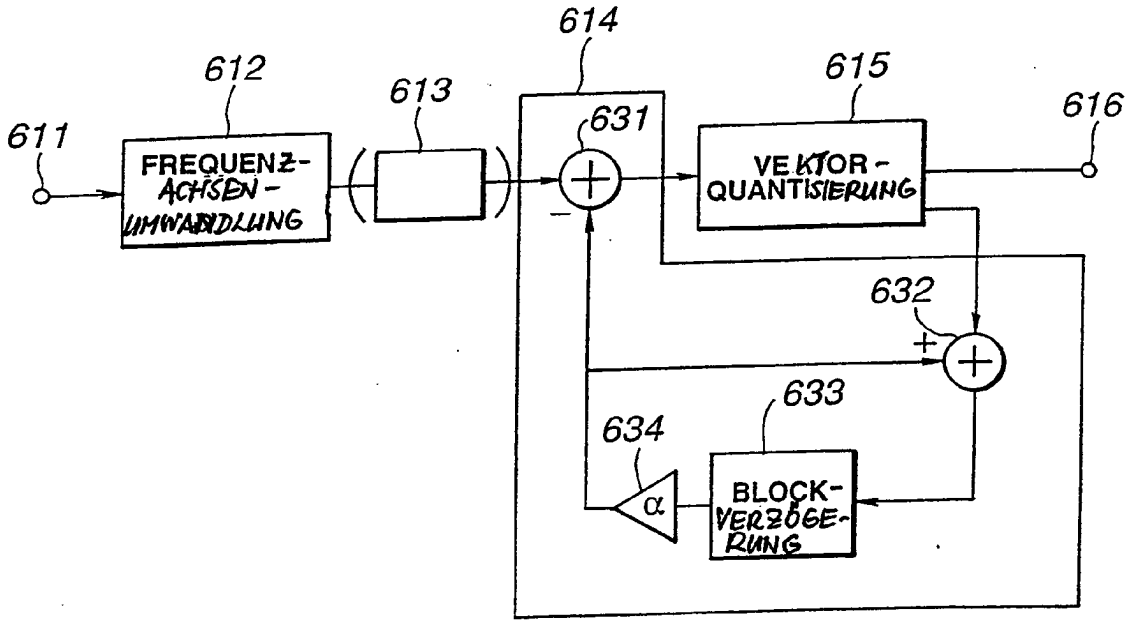

611
612
FREQUENZ-ACHSEN-UMWANDLUNG
613
614
631
615
VEKTOR-QUANTISIERUNG
616
632
633
634
α
BLOCK-VERZÖGERUNG

FIG.38

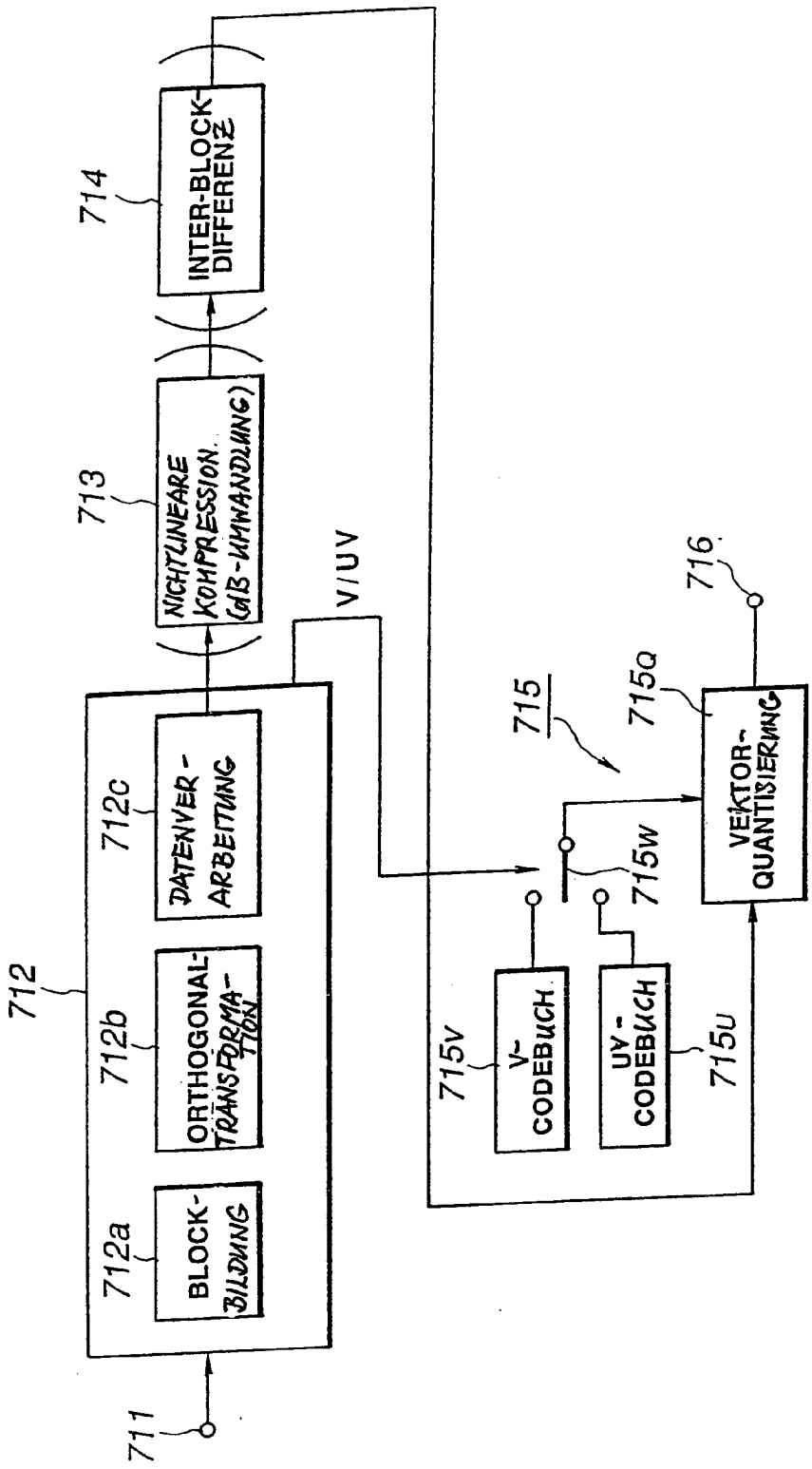

711
712
712a
BLOCK-BILDUNG
712b
ORTHOGONAL-TRANSFORMA-TION
712c
DATENVER-ARBEITUNG
713
NICHTLINEARE KOMPRESSION (A/B-UMWANDLUNG)
714
INTER-BLOCK-DIFFERENZ
V/UV
715
715V
V-CODEBUCH
715W
715U
UV-CODEBUCH
715Q
VEKTOR-QUANTISIERUNG
716

FIG.39

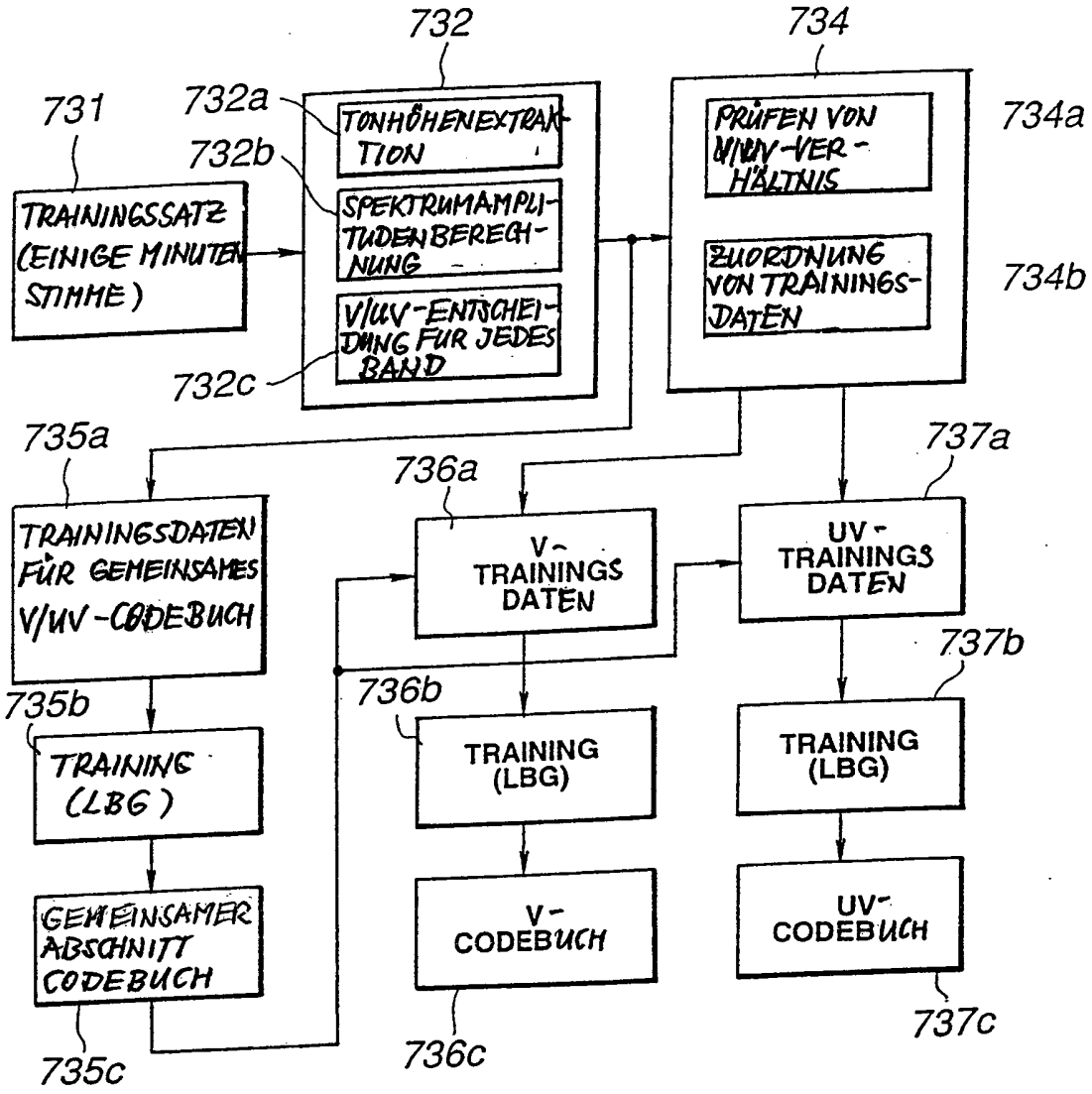

732
734
731
732a
732b
732c
734a
734b
TRAININGSSATZ (EINIGE MINUTEN STIMME)
TONHÖHENEXTRAKTION
SPEKTRUMAMPLITUDENBERECHNUNG
V/UV-ENTSCHEIDUNG FÜR JEDES BAND
PRÜFEN VON V/UV-VERHÄLTNIS
ZUORDNUNG VON TRAININGSDATEN
735a
736a
737a
TRAININGSDATEN FÜR GEMEINSAMES V/UV-CODEBUCH
V-TRAININGS DATEN
UV-TRAININGS DATEN
735b
736b
737b
TRAINING (LBG)
TRAINING (LBG)
TRAINING (LBG)
GEMEINSAMER ABSCHNITT CODEBUCH
V-CODEBUCH
UV-CODEBUCH
735c
736c
737c

FIG.40

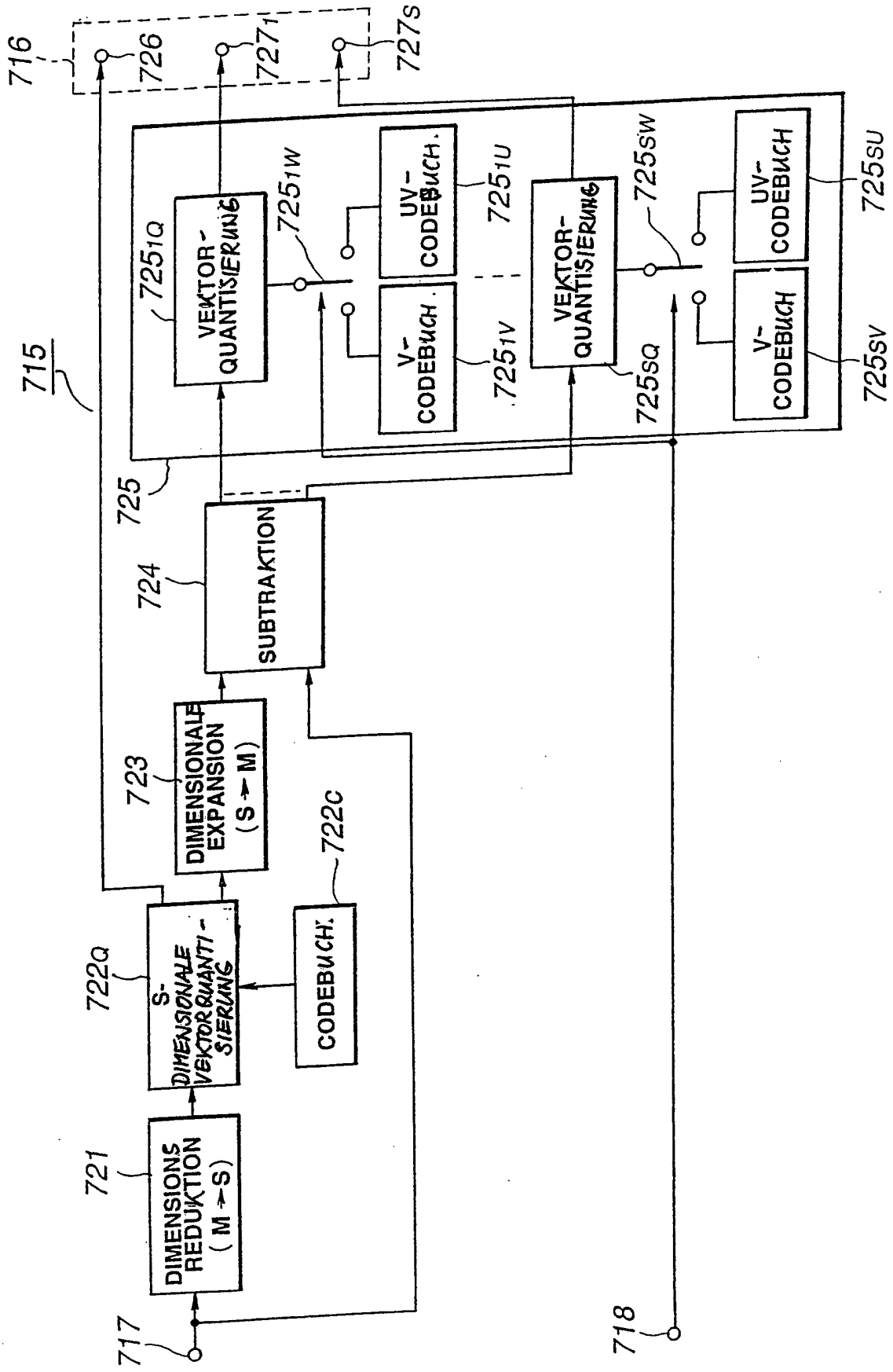

716
726
7271
727S
715
725
7251Q
VEKTOR-QUANTISIERUNG
7251W
UV-CODEBUCH
7251U
V-CODEBUCH
7251V
VEKTOR-QUANTISIERUNG
725SW
725SQ
UV-CODEBUCH
725SU
V-CODEBUCH
725SV
724
SUBTRAKTION
723
DIMENSIONALE EXPANSION (S→M)
722C
CODEBUCH
722Q
S-DIMENSIONALE VEKTORQUANTISIERUNG
721
DIMENSIONS REDUKTION (M→S)
717
718

FIG.41

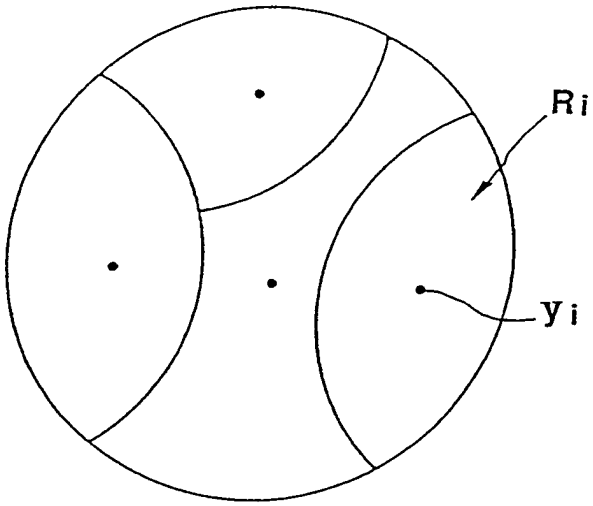

Ri
yi

FIG.42

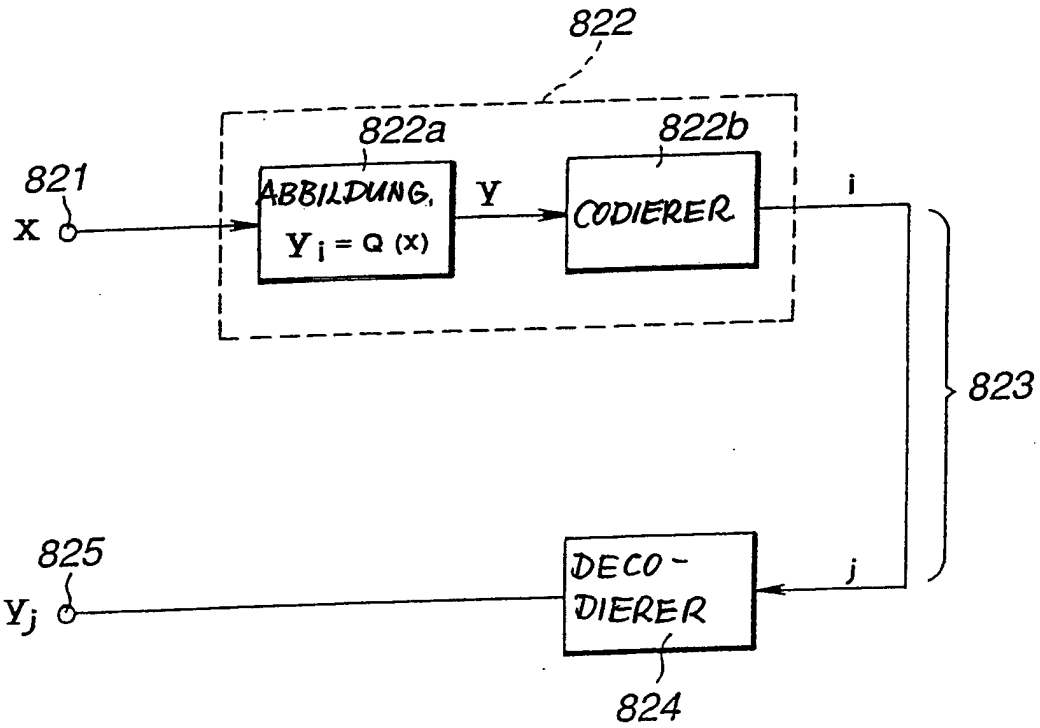

822
822a
822b
821
x
ABBILDUNG,
yi = Q (x)
y
CODIERER
i
823
825
yj
DECO-
DIERER
j
824

FIG.45

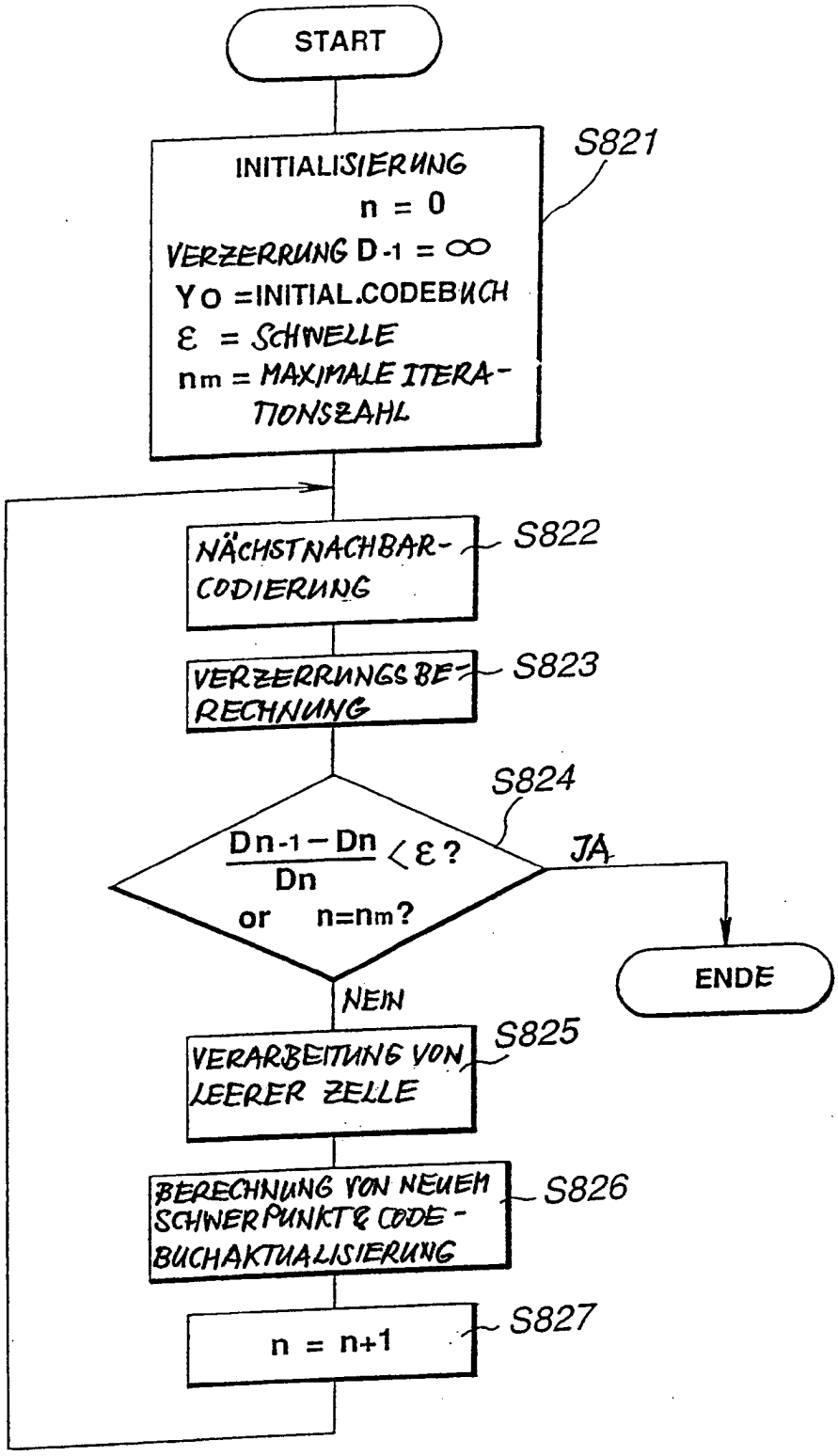

START
INITIALISIERUNG
n = 0
VERZERRUNG D-1 = ∞
Y0 =INITIAL.CODEBUCH
ε = SCHWELLE
nm = MAXIMALE ITERATIONSZAHL
S821
NÄCHSTNACHBAR-CODIERUNG
S822
VERZERRUNGSBERECHNUNG
S823
S824
Dn-1 − Dn / Dn < ε ?
or n=nm?
JA
ENDE
NEIN
VERARBEITUNG VON LEERER ZELLE
S825
BERECHNUNG VON NEUEM SCHWERPUNKT & CODEBUCHAKTUALISIERUNG
S826
n = n+1
S827

FIG.43

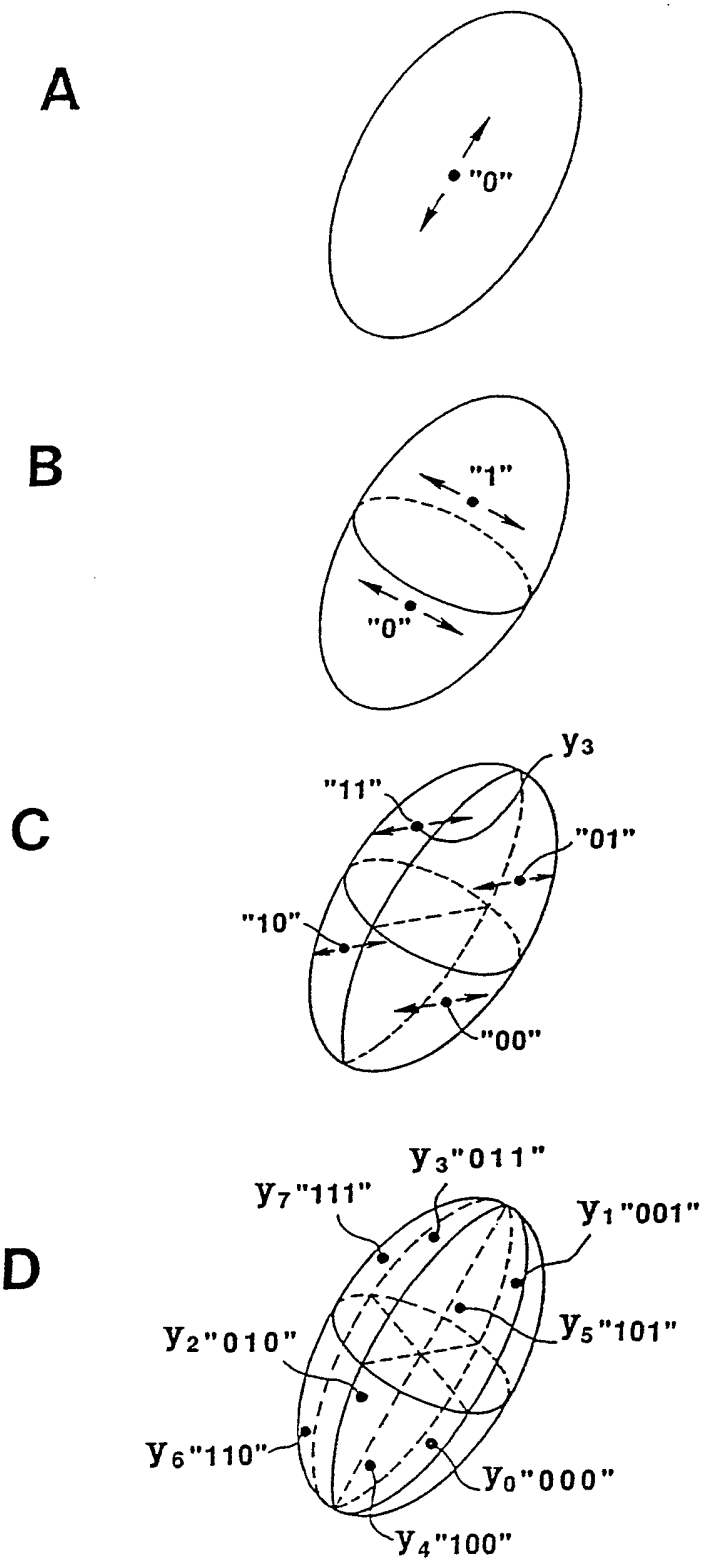

A
"0"
B
"1"
"0"
C
y3
"11"
"01"
"10"
"00"
D
y3"011"
y7"111"
y1"001"
y5"101"
y2"010"
y6"110"
y0"000"
y4"100"

FIG.44

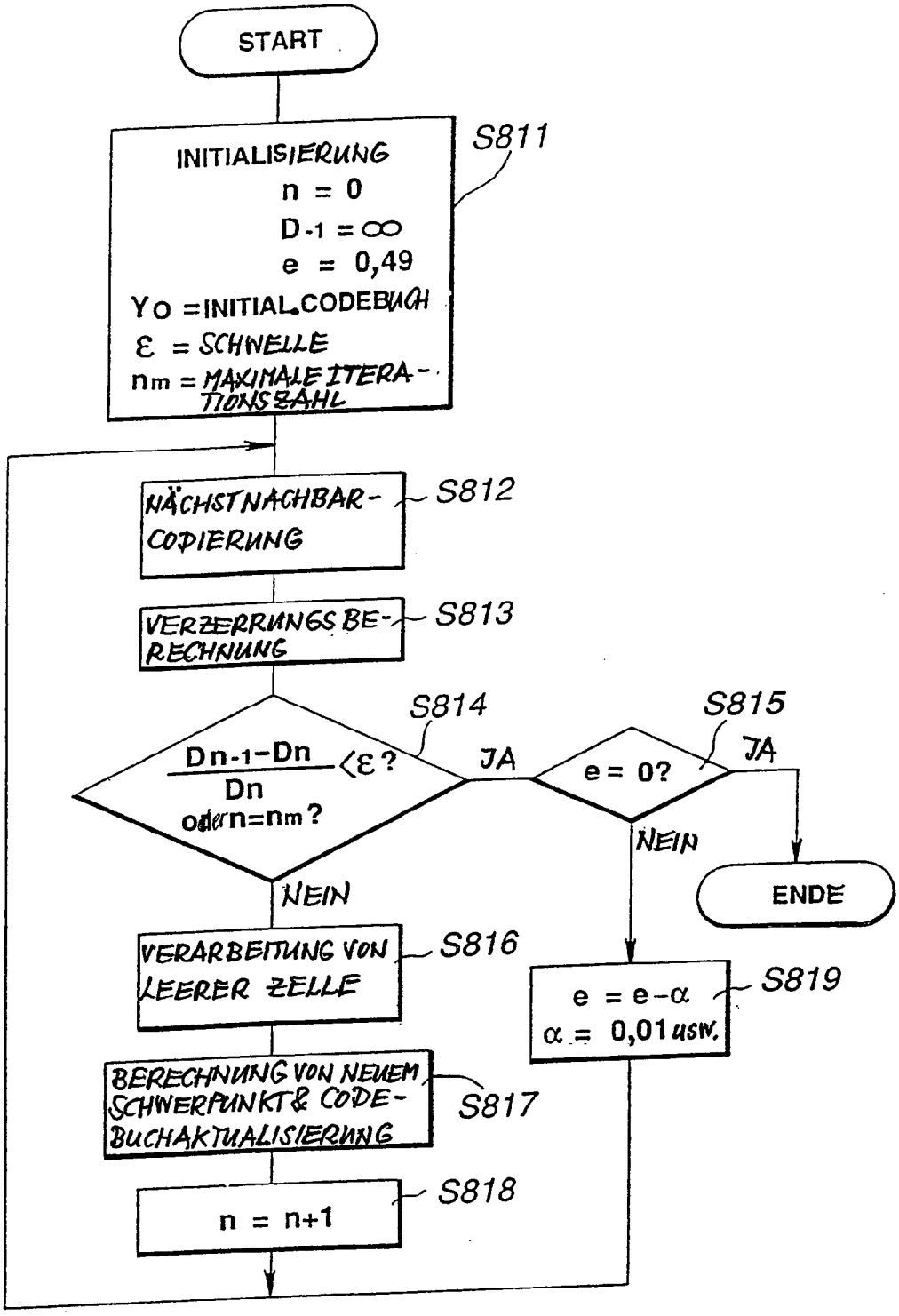

START
INITIALISIERUNG
n = 0
D-1 = ∞
e = 0,49
Yo = INITIAL CODEBUCH
ε = SCHWELLE
nm = MAXIMALE ITERATIONSZAHL
S811
NÄCHSTNACHBAR-CODIERUNG
S812
VERZERRUNGS BERECHNUNG
S813
Dn-1-Dn / Dn < ε ? oder n=nm ?
S814
JA
e = 0?
S815
JA
NEIN
NEIN
ENDE
VERARBEITUNG VON LEERER ZELLE
S816
e = e-α
α = 0,01 usw.
S819
BERECHNUNG VON NEUEM SCHWERPUNKT & CODEBUCHAKTUALISIERUNG
S817
n = n+1
S818

FIG.46

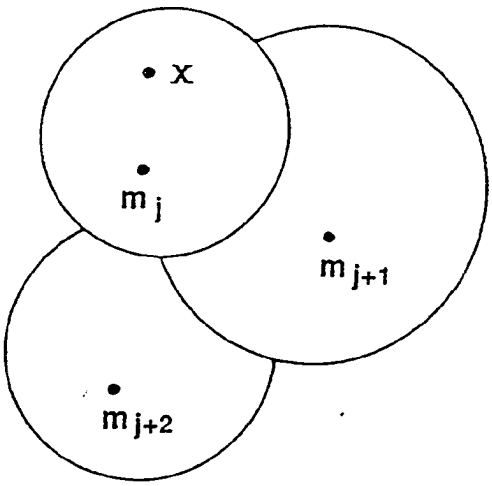
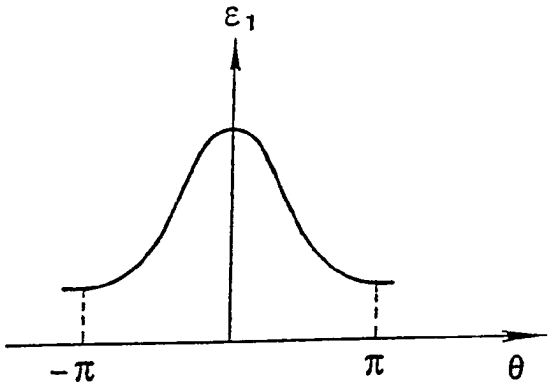


FIG.47

A



B

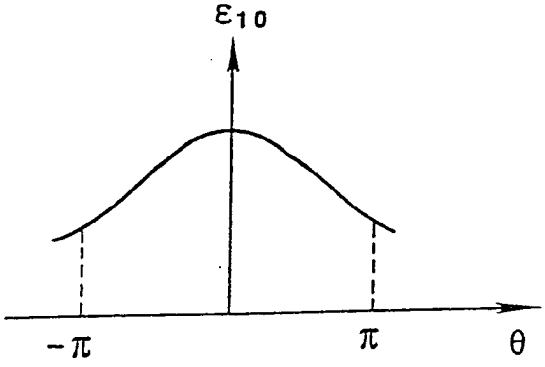


FIG.49

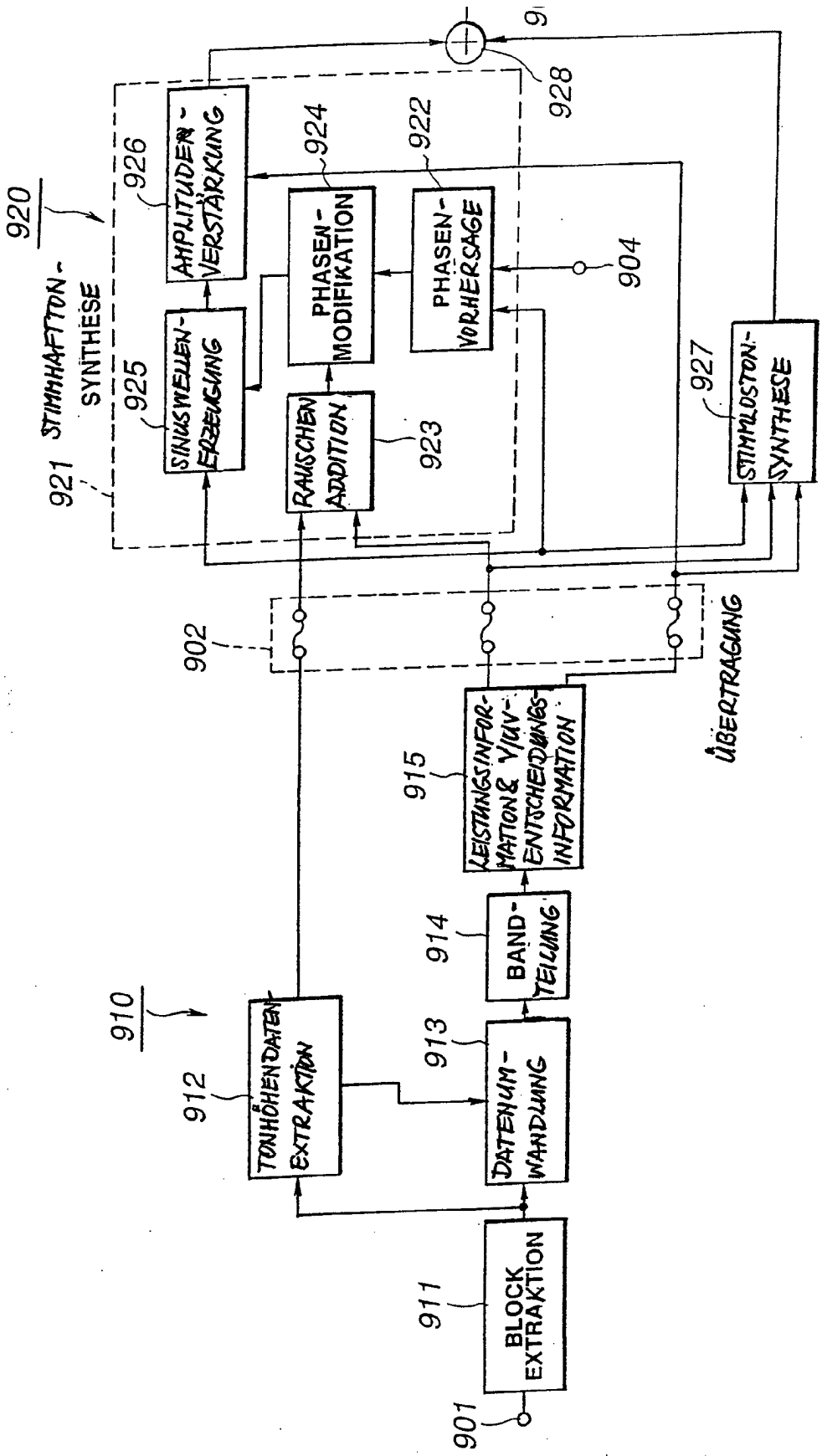

920
STIMMHAFTTON-SYNTHESE
921
926
AMPLITUDEN-VERSTÄRKUNG
924
PHASEN-MODIFIKATION
922
PHASEN-VORHERSAGE
904
925
SINUSWELLEN-ERZEUGUNG
RAUSCHEN ADDITION
923
927
STIMMLOSTON-SYNTHESE
928
902
915
LEISTUNGSINFORMATION & V/UV-ENTSCHEIDUNGS-INFORMATION
ÜBERTRAGUNG
910
914
BAND-TEILUNG
912
TONHÖHENDATEN EXTRAKTION
913
DATENUM-WANDLUNG
911
BLOCK EXTRAKTION
901

FIG.48