



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) **DE 699 19 185 T2** 2006.06.22

(12)

Übersetzung der europäischen Patentschrift

(97) **EP 1 116 333 B1**

(51) Int Cl.⁸: **H03M 3/02** (2006.01)

(21) Deutsches Aktenzeichen: **699 19 185.8**

(86) PCT-Aktenzeichen: **PCT/US99/22621**

(96) Europäisches Aktenzeichen: **99 956 503.9**

(87) PCT-Veröffentlichungs-Nr.: **WO 00/19615**

(86) PCT-Anmeldetag: **29.09.1999**

(87) Veröffentlichungstag

der PCT-Anmeldung: **06.04.2000**

(97) Erstveröffentlichung durch das EPA: **18.07.2001**

(97) Veröffentlichungstag

der Patenterteilung beim EPA: **04.08.2004**

(47) Veröffentlichungstag im Patentblatt: **22.06.2006**

(30) Unionspriorität:

163235 29.09.1998 US

(84) Benannte Vertragsstaaten:

**AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT,
LI, LU, MC, NL, PT, SE**

(73) Patentinhaber:

Cirrus Logic, Inc., Austin, Tex., US

(72) Erfinder:

MELANSON, Laurence, John, Boulder, US

(74) Vertreter:

**Müller-Boré & Partner, Patentanwälte, European
Patent Attorneys, 81671 München**

(54) Bezeichnung: **KORREKTUR NICHTLINEARER AUSGANGSVERZERRUNG IN EINEN DELTA-SIGMA D/A-WAND-
LER**

Anmerkung: Innerhalb von neun Monaten nach der Bekanntmachung des Hinweises auf die Erteilung des europäischen Patents kann jedermann beim Europäischen Patentamt gegen das erteilte europäische Patent Einspruch einlegen. Der Einspruch ist schriftlich einzureichen und zu begründen. Er gilt erst als eingelegt, wenn die Einspruchsgebühr entrichtet worden ist (Art. 99 (1) Europäisches Patentübereinkommen).

Die Übersetzung ist gemäß Artikel II § 3 Abs. 1 IntPatÜG 1991 vom Patentinhaber eingereicht worden. Sie wurde vom Deutschen Patent- und Markenamt inhaltlich nicht geprüft.

Beschreibung**Hintergrund der Erfindung****Gebiet der Erfindung:**

[0001] Die vorliegende Erfindung bezieht sich auf Digital/Analog-Konverter bzw. -Wandler bzw. -Umformer (DACs) für Tonfrequenzen. Genauer gesagt bezieht sich diese Erfindung auf die Korrektur einer nicht linearen Ausgabe- bzw. Ausgangsverzerrung in Delta-Sigma-DACs.

Beschreibung des Stands der Technik

[0002] Dokument WO 481.85, im Miteigentum des gegenwärtigen Anmelders, liefert den Hintergrund für die vorliegende Erfindung. Eine Delta-Sigma-Modulation gliedert eine Rausch-Gestaltungs-Technik ein, wodurch das Rauschen eines Quantisierers (häufig 1-bit), welcher auf einer Frequenz arbeitet, welche viel größer als die Bandbreite ist, zu Frequenzen bewegt wird, welche in dem Ausgabe- bzw. Ausgangssignal nicht von Interesse sind. Ein Filter nach bzw. hinter dem Quantisierer entfernt das Äußere des Bandrauschens. Das resultierende System synthetisiert einen hochauflösenden Datenkonverter, ist jedoch aus niedrig auflösenden Funktionsbausteinen konstruiert. Eine gute Übersicht über die Theorie von Delta-Sigma-Modulation wird in "Oversampling Delta-Sigma Data Converters", von Candy und Temes, IEEE Press, 1992 gegeben.

[0003] **Fig. 1** zeigt einen wohlbekannten Delta-Sigma-Quantisierer erster Ordnung. Der Zweck dieses Quantisierers, in einem Digital/Analog-(D/A)-Konverter, ist es, ein hochauflösendes Digitalsignal x_i , **11**, welches einige Bits (**16** beispielsweise) aufweist, in einen Einzel-Bit-Code y_i , **12** zu konvertieren bzw. zu wandeln, welcher genau zu analog gewandelt werden kann. Eine Eingabe **11** wird dem Quantisierer über einen Integrator **16** zugeführt und ein quantisierter Ausgang bzw. eine quantisierte Ausgabe **12** wird als Rückkopplung **25** rückgeführt und unter Verwendung eines Addierers **14** von der Eingabe subtrahiert. Integrator **16** umfaßt Verzögerung **18** und Addierer **17**, welcher das verzögerte Signal **19** zu dem Signal von dem Addierer **14** addiert. Der Quantisierer **21** wirkt als ein 1-Bit A/D-Konverter, welcher dessen Ausgabe bzw. Ausgang zu einer negativen Eins (eine negative Eingabe vorausgesetzt) oder einer positiven Eins (eine positive Eingabe vorausgesetzt) ansteuert. Die Quantisierungsfunktion wird geformt, indem der Ausgang bzw. die Ausgabe des Integrators **16** zu einem Fehlersignal e_i , **23** addiert wird. Dieses Formen bzw. Modellieren erlaubt es, die Berechnung des Spektrums des Rauschens in einer direkten Art und Weise zu erledigen.

[0004] Für große positive Eingaben wird die Integratorausgabe positiv sein. Eine logische Eins ist dann der Ausgang des Quantisierers, welche rückgekoppelt und von der Eingabe subtrahiert wird. Die Serie von ausgegebenen Einsen setzt sich fort, bis die Integratorausgabe, welche aufgrund der negativen Rückkopplung nach unten läuft, schließlich die Quantisierungsschwelle kreuzt, wenn der Quantisierer eine negative Eins ausgibt. Über die Zeit entspricht die Durchschnittsausgabe y_i der Eingabe x_i . Das System wird ein Delta-Sigma-Konverter- bzw. -Wandler erster Ordnung genannt, da eine einzige Integratorstufe verwendet wird.

[0005] **Fig. 2** zeigt, wie eine Ausgabe y_i , **31**, für eine sinusförmige Eingabe x_i , **30** aussieht. Wenn die Eingabe- bzw. Eingangs-Sinuswelle auf einem hohen Pegel ist, wird eine große Anzahl von Einsen generiert bzw. erzeugt. Wenn die Eingangs-Sinuswelle auf einem niedrigen Pegel ist, wird eine große Anzahl von negativen Einsen erzeugt.

[0006] **Fig. 3** zeigt einen üblichen Delta-Sigma-Quantisierer zweiter Ordnung. In der Praxis sind Delta-Sigma-Modulatoren im allgemeinen wenigstens zweiter Ordnung, da Modulatoren höherer Ordnung besser ein Rauschen in dem Signalband aufgrund einer verbesserten Vorhersage des bandinternen Quantisierungsfehlers reduzieren. Deshalb ist das resultierende Signal-Rausch-Verhältnis besser. Delta-Sigma-Modulatoren zweiter Ordnung sind noch relativ stabil und leicht zu entwerfen.

[0007] Ein Eingang x_i , **35** wird zu einem Rückkopplungssignal **54** durch einen Addierer **38** hinzuaddiert. Das Signal vom Addierer **38** wird einem ersten Akkumulator **40** zugeführt, welcher eine Verzögerung **42** und einen Addierer **41** umfaßt. Der Ausgang des Akkumulators **40** wird einem Rückkopplungssignal **54** hinzuaddiert und einem zweiten Akkumulator **44** zugeführt, welcher eine Verzögerung **47** und einen Addierer **45** umfaßt. Der Ausgang des Akkumulators **44** geht in einen Quantisierer **50**, gebildet als ein Fehlersignal e_i , **52**, hinzuaddiert zu dem Eingang durch einen Addierer **51**. Ein quantisierter Ausgang **36** wird auch ein Rückkopplungssignal **54** rückgeführt. Der Quantisierer **50** kann das Signal in Einsen und Nullen (1-Bit-Format) oder in Vielfachpegel quantisieren.

[0008] [Fig. 4](#) zeigt einen Oversampling-Digital/Analog-(D/A)-Konverter bzw. -Wandler, welcher einen Delta-Sigma-Quantisierer **70** zweiter Ordnung und einen Ein-Bit D/A-Konverter **71** wie den Demodulator **69** verwendet, und ein Tiefpaßfilter **73** zum Entfernen des Rauschens von dem 1-Bit-Signal. In einem bestimmten Beispiel des Oversampling-D/A-Konverters von [Fig. 4](#) besteht das Eingangssignal x_i , **60** aus Daten, welche in 16 Bit-Worte bei 8 kHz codiert wurden. Diese Worte werden in einem Register **63** plaziert, von welchem diese einem Tiefpaßfilter **64** bei 32 kHz zugeführt werden, wobei jedes Wort vier Mal wiederholt wird. Das Tiefpaßfilter würde typischerweise eines vom Typ mit endlicher Impulsantwort sein. Der lineare Interpolierer **66**, welcher auch ein Tiefpaßfilter ist, setzt drei neue Worte zwischen jedes Paar von Worten von dem Tiefpaßfilter **64** ein, welches die Datenrate auf 128 kHz steigert. Diese Worte werden in ein zweites Register **67** eingespeist, welches jedes Wort in den Demodulator **69** einspeist, wobei jedes Wort acht Mal wiederholt wird, was in einer Datenrate von 1 MHz resultiert. Dieses Wiederholen der Proben ist ein einfacher Typ eines Tiefpaßfilters. Die 1 MHz Abtastrate ist eine hinlänglich bzw. ausreichend hohe Datenrate, so daß das Quantisierungsrauschen, welches in das Signal eingebracht wird, klein ist, und die Anforderungen des analogen Glättungsfilters werden leicht erfüllt. Ein Ausgang y_i , **61** ist ein analoges Signal.

[0009] Techniken zum Steigern der Abtastrate, im allgemeinen Interpolation genannt, werden von in der Technik Versierten gut verstanden. Die meisten Entwürfe bzw. Entwicklungen werden verschiedene Stufen einer Steigerung verwenden, wobei jede folgende Stufe einfacher in der Struktur ist und bei einer schnelleren Rate bzw. Geschwindigkeit läuft.

[0010] Diese Art eines Demodulators wird häufig bei Audio-Applikationen angewendet. Der Ausgang des Demodulators **69** kann manchmal direkt in einen Lautsprecher (nicht gezeigt) gelenkt werden, da der Lautsprecher als ein Tiefpaßfilter wirken kann. Diese Konfiguration verwendet das, was ein Klasse-D-Ausgang genannt wird. Eine Verlustleistung in einer Klasse D-Stufe hat das Potential, sehr klein zu sein, da die Ausgangstransistoren immer in entweder einem vollständig kurzgeschlossenen oder offenen Position sind, wodurch der meiste Widerstandsleistungsverbrauch entfernt wird. Die verbleibende Leistung wird durch das Schalten einer Kapazität abgeführt, welche gleich ist zu $C \cdot V^2 \cdot F$. C, die zu schaltende Kapazität, wird typischerweise durch die parasitäre Kapazität des Ausgangsumformers und der Treibertransistoren eingestellt. V, die zu schaltende Spannung, wird durch die verfügbare Versorgung und den erforderlichen Audioausgang eingestellt. F, die Durchschnittsfrequenz des Ausgangs, kann durch den Entwickler variiert werden. Wenn F größer gemacht wird, verbessert sich die Qualität des Signals, jedoch steigt auch die Leistung an. Auch erfordern die Berechnungen selbst eine Verlustleistung.

[0011] Ein überabtastender bzw. Oversampling-Digital/Analog-(D/A)-Konverter, wie jener von [Fig. 4](#), welcher einen Quantisierer **70** der zweiten Ordnung und ein Tiefpaßfilter **71** zum Konvertieren der Daten von dem Delta-Sigma-Quantisierer **70** zum Analogsignal y_i , **61** verwendet, ist eine sehr effektive Vorrichtung. Jedoch weist sie eine relativ hohe Ausgangsdaten-Übergangsrate auf, welche eine höhere Leistung erfordert als wünschenswert ist. Die Verwendung eines Pulsbreitenmodulators zum Konvertieren der hoch auflösenden, niedrig bewerteten Daten in ein niedrig auflösendes, hoch bewertetes Ausgangssignal reduziert die Übergangsrate, führt jedoch eine nicht lineare Verzerrung in das Ausgangssignal ein, wodurch dessen Genauigkeit reduziert wird.

[0012] Das Europäische Patent EP 0576 701 B1 zeigt eine Hörhilfe, welche einen A/D-Konverter, um den Eingangsschall zu einem Digitalformat zu konvertieren, eine konventionelle digitale Signalverarbeitung, und einen Pulsbreiten-Modulator zum Konvertieren der hoch auflösenden, niedrig bewerteten bzw. niedrigfrequenten Daten in einen niedrig auflösenden, hoch bewerteten Klasse D-Ausgang aufweist, was bedeutet, daß der digitale Bitstrom einen Ausgangsumformer direkt antreibt. Der Ausgang des Pulsbreiten-Modulators weist relativ niedrige Übergangsraten auf, wobei jedoch, da es kein Rauschformen in dem Schaltkreis gibt, dieser ein niedriges Signal/Rausch-Verhältnis aufweist. Eine extrem hohe Taktrate würde erforderlich sein, um eine gute Audio-Wiedergabetreue zu erzielen.

[0013] Andere systemische nicht lineare Effekte können in das Ausgangssignal durch den ausgegebenen 1-Bit oder Vielfachpegel DAC oder andere Teile des Ausgangssystems eingebracht werden. Es ist wohl verstanden, daß ein Delta-Sigma-Konverter teilweise für eine Rückkopplung einer Nicht-Linearität korrigiert werden kann, indem ein nicht-lineares Element eingebracht wird, welches die gesamten Feedback- bzw. Rückkopplungspfade antreibt. Es gibt viele Arten von Ausgangsverzerrungen, welche nicht durch diese Mittel korrigiert werden können. Es verbleibt ein Bedarf für einen D/A-Konverter, welcher imstande ist, eine nicht-lineare Ausgangsverzerrung zu kompensieren, welche durch vorhersagbare systemische Faktoren verursacht werden, die das einzelne Element nicht korrigieren kann.

[0014] Ein Gegenstand bzw. Ziel der vorliegenden Erfindung ist es, eine systemische, nicht-lineare Verzerrung zu kompensieren, welche in das Ausgangssignal eines Delta-Sigma-Digital/Analog-Konverters (DAC) eingebracht wurde. Ein derartiger DAC wird als besonders nützlich für Audiosysteme empfunden, welche eine Klasse D-Ausgangsstufe und einen PWM-Treiber für hochgenaue DAC-Systeme verwenden, indem einige der schwierigen Fragen einer analogen Genauigkeit entfernt werden, und für ein digitales Korrigieren analoger Mangelhaftigkeiten in allgemeinen Delta-Sigma-Konvertern.

[0015] Daher wird in Übereinstimmung mit der Erfindung ein kompensierter Delta-Sigma-Digital/Analog-Konverter zur Verfügung gestellt, wie in den anhängigen Ansprüchen definiert wird.

[0016] Ein Beispiel ist ein DAC für eine Hörhilfe, wo eine Verlustleistung ernstlich begrenzt ist. Eine derartige Vorrichtung würde bestehen aus einem Mikrophon zum Empfangen von Umgebungsschall, einem Analog/Digital-Konverter zum Konvertieren des Schalls zu einem hoch auflösenden, niederfrequenten digitalen Signal, einer digitalen Verarbeitungsschaltanordnung zum Modifizieren der Qualität des hoch auflösenden, niederfrequenten digitalen Signals, einer Interpolatorschaltanordnung zum Konvertieren des hoch auflösenden mittelfrequenten, digitalen Signals zu einem mittelfrequenten, hochauflösenden Signal, und einem Digital/Analog-Konverter zum Konvertieren des modifizierten hoch auflösenden, mittelfrequenten digitalen Signals zurück in analogen Schall.

[0017] Der DAC konvertiert ein mittelfrequentes, hoch auflösendes digitales Signal in ein Analogsignal und beinhaltet einen Delta-Sigma-Modulator zum Konvertieren des mittelfrequenten, hochauflösenden digitalen Signals in ein mittelfrequentes, mittelauflösendes digitales Signal und kann einen Lastverhältnis-Demodulator beinhalten, welcher mit dem Delta-Sigma-Modulator zum Konvertieren des mittelfrequenten, mittelauflösenden digitalen Signals in ein hochfrequentes, hochauflösendes digitales Signal verbunden ist, und eine D/A-Vorrichtung, welche mit dem Lastverhältnis-Demodulator zum Konvertieren des hochfrequenten, niedrig auflösenden digitalen Signals in das Analogsignal verbunden ist. Der Lastverhältnis-Demodulator formatiert das hochfrequente, niedrig auflösende digitale Signal in ein vorbestimmtes Format mit niedriger Übergangsrate bzw. Frequenz.

[0018] Als eine Eigenschaft bzw. ein Merkmal ist der Delta-Sigma-Modulator ein Delta-Sigma-Modulator wenigstens zweiter Ordnung, welcher wenigstens zwei Feedback- bzw. Rückkopplungsschleifen aufweist, welche ein Feedback- bzw. Rückkopplungssignal tragen, und auch einen Korrekturfaktor (gespeichert beispielsweise in einem ROM) auswählt und an wenigstens eine der Rückkopplungsschleifen anwendet, basierend auf dem vorbestimmten Format niedriger Übergangsfrequenz und dem Feedback- bzw. Rückkopplungssignal.

[0019] Der Lastzyklus- bzw. Lastverhältnis-Demodulator kann Wechsellmittel einschließen, um das vorbestimmte Format mit niedriger Übergangsfrequenz bzw. -rate zwischen wenigstens zwei unterschiedlichen Formaten zu wechseln.

[0020] Der Delta-Sigma-Modulator kann auch ein Delta-Sigma-Modulator mit einer höheren als zweiter Ordnung sein, welcher mehr als zwei Feedback- bzw. Rückkopplungsschleifen aufweist, welche ein Feedback- bzw. Rückkopplungssignal tragen, und Korrekturfaktoren auswählen und an mehr als eine der Feedback- bzw. Rückkopplungsschleifen anwenden, basierend auf dem vorbestimmten Format mit niedriger Übergangsfrequenz und dem Rückkopplungssignal.

[0021] Der Lastverhältnis-Demodulator formatiert das hochfrequente, niedrig auflösende digitale Signal in ein vorbestimmtes Format mit niedriger Übergangsfrequenz zum Bestimmen eines multiplen bzw. vielfachen Bitwerts entsprechend jedem Eingangswert des mittelfrequenten, mittelauflösenden digitalen Signals und zum Ausgeben der bestimmten Werte als das hochfrequente, niedrig auflösende Digitalsignal. Beispielsweise kann der Lastverhältnis-Demodulator einen Wert entsprechend jedem möglichen Eingabe- bzw. Eingangswert des mittelfrequenten, mittelauflösenden digitalen Signals (beispielsweise in einem ROM) speichern, und den entsprechenden gespeicherten Wert für jeden mittelfrequenten, mittelauflösenden digitalen Signaleingang durchsuchen. Der Lastverhältnis-Demodulator kann einen Parallel-Seriell-Konverter bzw. Umwandler verwenden, welcher mit dem ROM verbunden ist, um die parallelen gesuchten Werte in das hochfrequente, niedrig auflösende niedrige Übergangs-Digitalsignal zu konvertieren.

[0022] Der DAC kann eine Klasse-D-Ausgangsstufe zum Antreiben eines Ausgangslautsprechers aufweisen. Besonders in dem Fall einer Hörhilfe kann diese in einer integrierten Schaltung implementiert sein und bei 1,1

Volt oder niedriger arbeiten.

[0023] Ein einzelner Multiphasenaddierer kann die gesamten arithmetischen Elemente des Akkumulators bzw. Speichers umfassen und arithmetische Elemente innerhalb des Delta-Sigma-Modulators rückkoppeln. Beispielsweise kann der einzelne Multiphasenaddierer einen Dreieingangs-Addierer umfassen, welcher in drei Phasen arbeitet. In diesem Fall addiert die erste Phase den Eingang zu dem Delta-Sigma-Modulator plus dem vorhergehenden Ausgang der ersten Stufe plus dem Negativen der Rückkopplung, die zweite Phase addiert den gegenwärtigen Ausgang der ersten Stufe plus dem vorhergehenden Ausgang der zweiten Stufe, und eine dritte Phase addiert den gegenwärtigen Ausgang der ersten Stufe plus den gegenwärtigen Ausgang der zweiten Stufe und speist den Quantisierer.

[0024] Die vorliegende Erfindung ist auch in vielen anderen Umgebungen nützlich bzw. verwendbar, um andere Typen einer nicht linearen Rückkopplungsverzerrung zu betrachten und zu korrigieren. Der Delta-Sigma-Modulator ist in jedem Fall wenigstens von zweiter Ordnung, welcher Rückkopplungsschleifen zu jedem Akkumulator bzw. Speicher aufweist. Jede Rückkopplungsschleife weist einen unterschiedlichen Korrekturfaktor auf, welcher an diese angewendet wird, um wiederholbare nicht lineare Verzerrungen in den Ausgangsstufen zu kompensieren. Diese Korrekturfaktoren unterscheiden sich voneinander auf nicht triviale Weise, um eine nicht lineare Verzerrung zu berücksichtigen.

[0025] Ein zweites Beispiel einer Umgebung, in welcher ein Delta-Sigma-Modulator einschließlich einer Rückkopplungskorrektur gemäß der vorliegenden Erfindung nützlich ist, ist, wo der Delta-Sigma-Modulator einschließlich der Korrekturschaltanordnung mit einem 1-Bit oder Multilevel- bzw. Vielfachpegel-Ausgangs DAC verbunden ist. Der DAC kann Ausgangspulse generieren bzw. erzeugen, welche Probleme mit der Anstiegsgeschwindigkeit, Variationen in der Dauer oder Höhe der Pulse, oder andere nicht lineare systemische Ausgangssignalfehler aufweisen.

[0026] Ein Delta-Sigma-Modulator, einschließlich einer Rückkopplungskorrektur gemäß der vorliegenden Erfindung ist auch zum Korrigieren von Fehlern nützlich bzw. verwendbar, welche in einem hochqualitativen Multilevel-Digital/Analog-Konverter erzeugt werden, welcher einen Mustergenerator und DACs mit Vielfachausgang verwendet. Jeder DAC neigt zum Erzeugen von Pulsen, welche nicht lineare systemische Variationen bzw. Abweichungen aufweisen.

[0027] Ein Delta-Sigma-Modulator, einschließlich einer Rückkopplungskorrektur gemäß der vorliegenden Erfindung ist auch zur Korrektur von Fehlern nützlich, welche in einem Audio-Leistungsverstärker erzeugt werden, welcher einen Lautsprecher in einem Klasse-D-Modus antreibt. Die von den Klasse-D-Ausgangstreibern erzeugten Signale weisen Probleme mit der Anstiegsgeschwindigkeit auf, was zu nicht linearen Fehlern in dem Antriebs- bzw. Ansteuersignal führt.

Kurze Beschreibung der Zeichnungen

[0028] [Fig. 1](#) (Stand der Technik) zeigt einen gut bekannten Delta-Sigma-Quantisierer der ersten Ordnung.

[0029] [Fig. 2](#) zeigt einen Ausgang y_i für einen sinusförmigen Eingang x_i für den Quantisierer von [Fig. 1](#).

[0030] [Fig. 3](#) (Stand der Technik) zeigt einen üblichen bzw. gemeinsamen Delta-Sigma-Quantisierer der zweiten Ordnung.

[0031] [Fig. 4](#) (Stand der Technik) zeigt ein Oversampling- bzw. Überabtast-D/A-Konvertersystem, welches einen Delta-Sigma-Quantisierer der zweiten Ordnung wie jenen von [Fig. 3](#) und einen 1-Bit D/A-Konverter als den Demodulator verwendet.

[0032] [Fig. 5](#) zeigt einen Demodulator gemäß der vorliegenden Erfindung, einschließlich eines Delta-Sigma-Datenkonverters und eines Lastverhältnis-Demodulators.

[0033] [Fig. 6](#) zeigt ein detaillierteres Diagramm des Lastverhältnis-Demodulators von [Fig. 5](#).

[0034] [Fig. 7](#) ist aus [Fig. 7A](#) bis [Fig. 7E](#) zusammengesetzt und zeigt verschiedene nützliche Datenformate zum Ausgeben von dem Lastverhältnis-Demodulator von [Fig. 6](#). [Fig. 7A](#) zeigt das zentrierte, nach rechts anwachsende Format. [Fig. 7B](#) zeigt das zentrierte, nach links anwachsende Format. [Fig. 7C](#) zeigt das Anwachsen von dem rechten Format. [Fig. 7D](#) zeigt das Anwachsen bzw. Ansteigen von dem linken Format. [Fig. 7E](#)

zeigt ein wechselndes Format, worin ungerade Zyklen bzw. Takte von rechts anwachsen, und gerade Zyklen bzw. Takte von links anwachsen.

[0035] [Fig. 8](#) zeigt einen Delta-Sigma-Quantisierer der zweiten Ordnung, welcher für Niedrigübergangs-De-modulator-Ausgangsdaten rückkopplungs-kompensiert ist, gemäß einer ersten Ausführungsform der vorliegenden Erfindung.

[0036] [Fig. 9](#) zeigt einen anderen kompensierten Delta-Sigma-Quantisierer der zweiten Ordnung gemäß einer zweiten Ausführungsform der vorliegenden Erfindung.

[0037] [Fig. 10](#) zeigt einen Demodulator gemäß der vorliegenden Erfindung, welcher einen dritten kompensierten Delta-Sigma-Quantisierer der zweiten Ordnung verwendet.

[0038] [Fig. 11](#) zeigt einen Signalflußgraph eines Delta-Sigma-Quantisierers, welcher in [Fig. 10](#) verwendet werden könnte.

[0039] [Fig. 12](#) zeigt einen kompensierten Delta-Sigma-Quantisierer der dritten Ordnung gemäß einer vierten Ausführungsform der vorliegenden Erfindung.

[0040] [Fig. 13](#) zeigt das Ausgangssignal des Demodulators von [Fig. 5](#), welcher einen Delta-Sigma-Quantisierer ohne die Kompensation der vorliegenden Erfindung verwendet, für ein Eingangssignal, welches eine Sinuswelle bei -40 dB umfaßt.

[0041] [Fig. 14](#) zeigt das Ausgangssignal des Demodulators von [Fig. 5](#), welcher den kompensierten Delta-Sigma-Quantisierer von [Fig. 10](#) verwendet, für ein Eingangssignal, welches eine Sinuswelle bei -40 dB umfaßt.

[0042] [Fig. 15](#) zeigt das Ausgangssignal des Demodulators von [Fig. 5](#), welcher einen Delta-Sigma-Quantisierer ohne die Kompensation der vorliegenden Erfindung verwendet, für ein Eingangssignal, welches eine Sinuswelle bei -60 dB umfaßt.

[0043] [Fig. 16](#) zeigt das Ausgangssignal des Demodulators von [Fig. 5](#), welcher den kompensierten Delta-Sigma-Quantisierer von [Fig. 10](#) einsetzt, für ein Eingangssignal, welches eine Sinuswelle bei -60 dB umfaßt.

[0044] [Fig. 17](#) stellt eine C-Programm-Simulation des Demodulators von [Fig. 5](#) zur Verfügung, welcher den kompensierten Delta-Sigma-Quantisierer von [Fig. 10](#) verwendet.

[0045] [Fig. 18](#) zeigt eine Hörhilfe, welche den Demodulator von [Fig. 5](#) verwendet.

[0046] [Fig. 19](#) zeigt einen Delta-Sigma-Modulator der zweiten Ordnung, welcher eine Feedback- bzw. Rückkopplung für nichtlineare Abweichungen der Ausgangsdaten kompensiert aufweist, gemäß einer sechsten Ausführungsform der vorliegenden Erfindung.

[0047] [Fig. 20](#) zeigt eine Serie von Ausgangspulsen, welche aus der Ausführungsform von [Fig. 19](#) resultieren, welche unterschiedliche Anstiegszeiten und Flächen aufweisen.

[0048] [Fig. 21](#) zeigt einen Puls einer Höhe zwei, resultierend aus der Multilevel-Version der Ausführungsform von [Fig. 19](#), welche illustriert, daß der Puls nicht dasselbe Schwerkraftzentrum oder die doppelte Fläche eines Einzelpulses aufweist.

[0049] [Fig. 22](#) zeigt eine Serie von Ausgangspulsen, die aus der Ausführungsform von [Fig. 19](#) resultieren, welche unterschiedliche Anstiegszeiten und Flächen aufweisen.

[0050] [Fig. 23](#) zeigt einen hochqualitativen Multilevel-DAC, welcher den kompensierten Delta-Sigma-Modulator von [Fig. 19](#) verwendet.

[0051] [Fig. 24](#) zeigt einen Klasse-D-Leistungsverstärker, welcher den kompensierten Delta-Sigma-Modulator von [Fig. 19](#) verwendet.

[0052] [Fig. 25](#) zeigt ein Zeitdiagramm der Ausführungsform von [Fig. 24](#).

Detaillierte Beschreibung der bevorzugten Ausführungsform

[0053] [Fig. 5](#) zeigt einen Demodulator, welcher in einem Over-Sampling D/A-Konverter wie jenem verwendet werden könnte, der in [Fig. 4](#) gezeigt ist, welche einen Demodulator **69** in dieser Figur ersetzt. Hoch auflösende Daten **202**, beispielsweise 12 bis 20 Bit-Daten, treten in den Delta-Sigma-Konverter **204** ein. Die Abtastrate bzw. Abtastfrequenz dieser Daten wurde bereits von dem für die Codierung der Daten erforderlichen niederfrequenten Takt auf einen mittelfrequenten Takt erhöht, welcher zum Takten des Delta-Sigma-Konverters verwendet wird. Das Verhältnis des niedrigen zu dem mittleren Takt wird typischerweise ein Faktor von 32 bis 1024 sein, beispielsweise ein niedriger Takt von 16 kHz zu einem mittleren Takt von 1 MHz. Der Delta-Sigma-Modulator **204** wird durch einen mittleren Takt **213**, beispielsweise bei 1 MHz getaktet, um Daten **206** mittlerer Auflösung (2 bis 5 Bit beispielsweise) zu erzeugen. Ein Lastverhältnis-Demodulator **208** wird durch mittleren Takt **213** und hohen Takt **212** getaktet. Die Frequenz des hohen Takts ist ein Vielfaches des mittleren Takts, beispielsweise 16 MHz. Die Lastzyklus- bzw. Lastverhältnis-Demodulator **208** ist detaillierter in [Fig. 6](#) gezeigt. Der Ausgang des Lastverhältnis-Demodulators **208** sind niedrig auflösende Daten **210**, typischerweise in einem Ein- oder Zwei-Bit-Format, bei einer hohen Taktrate bzw. Taktfrequenz.

[0054] Der optionale 0,5 Mitteltakt **214** wird zum Wechseln von Ausgangsdatenformaten verwendet, wie beispielsweise das Wechseln des Ausgangsdatenformats, welches in [Fig. 7E](#) unten gezeigt ist. Da zwei unterschiedliche Ausgabe- bzw. Ausgangsformate auf abwechselnde Art und Weise verwendet werden, wählt die 0,5 Mitteltaktfrequenz bzw. -rate eines der Formate für jeden anderen Datenrahmenausgang von ROM **220** (gezeigt in [Fig. 6](#)) aus. Beispielsweise würde für jeden geraden Rahmen **0,5** Mitteltakt **214** niedrig sein, was dem ROM **220** sagt, das Anwachsen des rechten Formats zu verwenden. Für ungerade Rahmen wäre **0,5** Mitteltakt **214** hoch und das Anwachsen vom linken Format würde verwendet werden. **0,5** Mitteltakt **214** wird nicht verwendet für die in [Fig. 7A–D](#) gezeigten Datenformate, da diese Formate nicht abwechseln. ROM **220** muß zweimal so groß für den alternierenden Fall sein, um beide Formate zu halten bzw. aufzunehmen (Anwachsen von rechts und Anwachsen von links in dem 7E-Fall). Der Delta-Sigma-Modulator **204** verwendet auch **0,5** Mitteltakt **214** für den alternierenden Fall, da ein verschiedener Korrekturfaktor abhängig davon verwendet werden wird, welches Ausgangsformat durch ROM **220** angewendet wird.

[0055] [Fig. 6](#) zeigt den Lastverhältnis-Demodulator **208** in größerem Detail. Der Lastverhältnis-Demodulator **208** umfaßt nur kombinatorische Logik und beinhaltet keine Feedback- bzw. Rückkopplungsschleifen. Somit ist er deterministisch und weist wenig oder keinen Speicher (trägt minimalen bzw. Minimalzustand) auf. In der bevorzugten Ausführungsform umfaßt der Lastverhältnis-Demodulator **208** ein ROM **220**, beinhaltend eine Suchtabelle, und einen parallel/seriell Datenkonverter **223**. Eingangsdaten **206** mittlerer Auflösung werden verwendet, um Daten in ROM **220** zu suchen, welche parallel über Leitungen **221** ausgegeben werden. Die im ROM **220** gespeicherten Daten werden gewählt, um die Anzahl von Übergängen in seinem Ausgang zu minimieren, wodurch der Betrag der benötigten Leistung zum Antreiben des Demodulators minimiert wird. Abhängig davon, was im ROM **220** gespeichert ist, können die Daten auf den Leitungen **221** in einer Vielfalt von nützlichen Formaten mit niedriger Übergangsfrequenz bzw. Übergangsrate vorliegen, wie jene, die in [Fig. 7A–Fig. 7E](#) gezeigt sind. Das Verhältnis zwischen hohem Takt **212** und mittlerem Takt **213** ist durch die Anzahl von parallelen Leitungen **221** bestimmt. Beispielsweise kann, wenn es 16 parallele Leitungen **221** gibt, der hohe Takt **212** auf 16 MHz laufen und der Mitteltakt **213** kann auf 1 MHz laufen. Der Parallel/Seriell-Datenkonverter **223** würde 16 Bit von Daten parallel bei 1 MHz empfangen und serielle Daten auf 16 MHz ausgeben. Der hohe Takt **212** ist nur erforderlich, um den Parallel/Seriell-Konverter **223** zu betreiben.

[0056] Jene mit Erfahrung in der Technik werden anerkennen, daß viele gut bekannte Strukturen verwendet werden könnten, um den Lastverhältnis-Demodulator **208** zu implementieren. Beispielsweise könnte die Suchfunktion mit Gattern bzw. Gates anstelle des ROM **220** erledigt werden. Oder die Daten könnten in zwei Teilen zu je acht Bit beispielsweise ausgetaktet werden. Die Parallel-zu-Seriell-Konversion bzw. Umwandlung könnte in verschiedenen Stufen erledigt werden. Ihr Ausgang bzw. ihre Ausgabe könnte eher zwei parallele Bits als eines sein. Der Lastverhältnis-Demodulator könnte auch durch logische Decodierer realisiert werden, welche an den Ausgang des Frequenzteilers angewendet werden, welcher den Mitteltakt aus dem hohen Takt erzeugt. Zustands-Maschinenimplementierungen könnten auch verwendet werden.

[0057] Ein Pulsbreitenmodulator ist ein Beispiel eines möglichen Lastzyklus- bzw. Tastverhältnis-Demodulators **208**. In diesem Fall, würde der Inhalt von ROM **220** das Anwachsen vom linken Muster implementieren, welches in [Fig. 7D](#) gezeigt ist.

[0058] Fig. 7 ist aus Fig. 7A bis Fig. 7E zusammengesetzt und zeigt verschiedene mögliche Datenformatausgaben des Lastverhältnis-Demodulators 200 von Fig. 6. Fig. 7A zeigt das zentrierte, nach rechts anwachsende Format. Daher ist eine Eins rechts von der Mittellinie und eine Drei weist zwei Bits nach rechts und ein Bit nach links auf. Die geraden Zahlenwerte sind zentriert. Fig. 7B zeigt das zentrierte, nach links anwachsende Format. Daher ist eine Eins links von der Mittellinie und eine Drei weist zwei Bits nach links und ein Bit nach rechts auf.

[0059] Fig. 7C zeigt das Anwachsen des rechten Formats. Fig. 7D zeigt das von links anwachsende Format. Das von links und von rechts anwachsende Format resultieren in größerem Diskontinuitätsrauschen bzw. Unstetigkeitsrauschen als die anderen Formate und sind so etwas weniger nützlich.

[0060] Fig. 7E das abwechselnde Format, in welchem ungerade Zyklen bzw. Takte von rechts anwachsen, und gerade Zyklen bzw. Takte von links anwachsen. Dieses Format ist besonders nützlich, da es halb so viele Ausgangsdatenübergänge erfordert wie jedes der obigen Formate. Jedoch ist die Entwicklung komplexer, wie oben beschrieben wurde. Um dieses Format darzustellen bzw. zu illustrieren, werden die folgenden Werte aufeinanderfolgend gezeigt: 3, 2, 5, 1, 0, 3, 16, 4. Offensichtlich wäre es gleichwertig, ungerade Zyklen von links anwachsend zu haben, und gerade Zyklen von rechts anwachsend.

[0061] In all den in Fig. 7A–Fig. 7E gezeigten Formaten ist die Fläche unter der Ausgangskurve (erstes Integral) direkt proportional zum Eingangswert. Mit anderen Worten ist der Effekt erster Ordnung gleichwertig zu einer idealen Impulsfunktion. Das ist jedoch für das zweite Integral nicht wahr bzw. zutreffend, welches ein gewisses Niveau an Verzerrung hervorrufen wird. Eine einfache Beschreibung ist, daß der Schwerpunkt des Signals nicht an derselben Zeitposition in dem Signal verbleibt. Eine Compensation zum Erreichen niedriger Verzerrung und einem guten Signal-Rausch-Verhältnis ist erforderlich.

[0062] Fig. 8 zeigt eine erste Ausführungsform eines Delta-Sigma-Quantisierers der zweiten Ordnung, welcher als Delta-Sigma-Konverter 204 des Demodulators von Fig. 5 verwendet werden könnte. Der Delta-Sigma-Konverter von Fig. 8 weist ein Feedback bzw. eine Rückkopplung 92 auf, die (das) für das an die Daten angewendete Format durch den Lastverhältnis-Demodulator 208 kompensiert ist. Die Struktur der Ausführungsform von Fig. 8 ist identisch zur Struktur des Delta-Sigma-Quantisierers der zweiten Ordnung von Fig. 3, ausgenommen, daß die Rückkopplung in den Addierer 95 durch den Korrekturblock 93 korrigiert wurde und Quantisierer 90 ein Vielfach-Bit-Quantisierer ist.

[0063] Eingang xi, 78 wird zum Feedback- bzw. Rückkopplungssignal 92 durch den Addierer 80 hinzuaddiert. Das Signal vom Addierer 80 wird in den ersten Akkumulator bzw. Speicher 82 eingespeist, welcher eine Verzögerung 84 und einen Addierer 83 umfaßt. Der Ausgang des Akkumulators 82 wird zum korrigierten Rückkopplungssignal 94 hinzuaddiert und in den zweiten Akkumulator 86 eingespeist, welcher eine Verzögerung 88 und einen Addierer 87 umfaßt. Der Ausgang des Akkumulators 86 geht in den Multi-Bit-Quantisierer 90. Der quantisierte Ausgang 79 (yi) wird als Rückkopplungssignal 92 rückgekoppelt. Der Quantisierer 90 weist mehr als zwei Signalpegel für seinen Ausgang bzw. seine Ausgabe auf.

[0064] Der Korrekturblock 93 ist als eine Black Box gezeigt, welche das Feedback- bzw. Rückkopplungssignal 92 als einen Eingang und das Korrektursignal 94 als einen Ausgang aufweist. In der bevorzugten Ausführungsform verwendet der Korrekturblock 93 das Signal 92, um einen Korrekturfaktor im ROM zu suchen, und liefert ihn zum Addierer 95, um zum Feedback- bzw. Rückkopplungssignal 96 hinzu addiert zu werden. Äquivalent könnte der Korrekturblock als seinen Ausgang ein Signal aufweisen, welches das zum Korrekturfaktor hinzuaddierte Feedback- bzw. Rückkopplungssignal 92 umfaßt. Dann würde das Feedback- bzw. Rückkopplungssignal 96 entfernt werden.

[0065] Die Inhalte des ROM des Korrekturblocks 93 werden ausgewählt, um das Format des Datenausgangs von dem Lastverhältnis-Demodulator 208 zu korrigieren. Der erforderliche Ausgang des Korrekturblocks kann einfach berechnet werden, indem das zweite Integral des ausgegebenen bzw. Ausgangsdatenmusters genommen wird und der ideale, lineare Wert subtrahiert wird. Fig. 13 zeigt eine C Programm-Simulation eines Demodulators, welcher das zentrierte, nach rechts anwachsende Format von Fig. 7A verwendet. Das in diesem Programm verwendete ROM (welches diese Auswahl von Parametern, Datenformaten, Taktverhältnissen, usw. verwendet) würde wie folgt entworfen werden:

- 1) Wähle eine 0-Zeit-Position. In diesem Fall werden wir das Zentrum des Zeitrahmens als 0 verwenden.
- 2) Berechne die Zeiten des Zentrums von jedem einzelnen Einheitspuls. In dem vorliegenden Beispiel sind diese (von links nach rechts):
–15/32

-13/32
 -11/32
 -9/32
 -7/32
 -5/32
 -3/32
 -1/32
 1/32
 3/32
 5/32
 7/32
 9/32
 11/32
 13/32
 15/32

3) Summiere die Elemente von der obigen Berechnung, welche innerhalb eines beliebigen vorgegebenen Pulses verwendet werden. Als ein Beispiel wird für den Fall einer Breite 1 1/32 verwendet und die Summe ist 1/32. Für den Fall einer Breite 2 werden -1/32, 1/32 verwendet, und die Summe ist 0. Für den Fall der Breite drei werden -1/32, 1/32, 3/32 verwendet, und der Wert ist 3/32. Für den Fall der Breite fünf werden -3/32, -1/32, 1/32, 3/32, 5/32 verwendet, und der Wert ist 5/32. Dieser Wert wird mit dem Feedback- bzw. Rückkopplungswert für einen einzelnen Schritt multipliziert, in dem Fall des C-Programms von **Fig. 17**, 2048 (hex 0800), um den Korrekturtabellenwert **320** für die Eingabe **5** (hex0140) zu bekommen.

[0066] Da diese Pulse nach rechts anwachsen, sind die ungeraden Ausgaben bzw. Ausgänge relativ zum Zentrum spät. Daher wird der Korrekturtermzeichen entgegengesetzt zu dem Feedback-Rückkopplungsterm aufweisen.

[0067] Es ist auch möglich, einen Korrekturfaktor in jeder Feedback- bzw. Rückkopplungsschleife des Delta-Sigma-Konverters von **Fig. 8** (d.h. kompensiert den Pfad zum Addierer **80** ebenso wie zum Addierer **95**) zu verwenden. Diese Technik würde verwendet werden, wenn die Ausgangsstufe in der Energie nicht linear wäre, jedoch in normalen Fällen würde dieser Kompensationspfad eine eins-zu-eins Funktion (d.h. unkompensiert) sein.

[0068] **Fig. 9** zeigt einen anderen kompensierten Delta-Sigma-Quantisierer gemäß der zweiten Ordnung gemäß einer zweiten Ausführungsform der vorliegenden Erfindung. Dieser wird gezeigt und beschrieben, um darzustellen bzw. zu illustrieren, wie die Kompensationsfunktion in jedem Delta-Sigma-Quantisierer verwendet werden kann.

[0069] In dem kompensierten Delta-Sigma-Quantisierer von **Fig. 9** wird das Eingangssignal xi, **98** zu dem Negativen des Feedback- bzw. Rückkopplungssignals **111** durch einen Addierer **100** hinzuaddiert, um ein Signal **101** zu bilden. Ein Addierer **103** addiert **101** zu dem Signal von der Verzögerung **104** und speist das Resultat in den Addierer **106** ein. Der Addierer **106** addiert dieses Signal zu dem Negativen von Signal **113**, welches als Feedback bzw. Rückkopplung **111** mit 2 multipliziert wird, und zum Signal **116**, welches der Korrekturterm ist, welcher durch den Korrekturblock **114** gebildet wurde. Der Ausgang des Addierers **106** wird zu dem Signal von der Verzögerung **108** durch den Addierer **107** hinzu addiert. Das Resultat wird durch den Quantisierer **110** quantisiert, um die Ausgabe **99** und das Feedback- bzw. Rückkopplungssignal **111** zu bilden. Feedback **111** wird für den variablen Schwerpunkt durch den Korrekturblock **114** kompensiert, wie dies oben beschrieben wurde.

[0070] **Fig. 10** zeigt einen Demodulator in Übereinstimmung mit der vorliegenden Erfindung, welcher einen kompensierten Delta-Sigma-Konverter **204a** zweiter Ordnung verwendet. Diese besondere Implementierung ist besonders zur Verwendung mit einem Tastverhältnis bzw. Lastzyklus geeignet, da die Berechnungen in multiplen bzw. mehreren Phasen durchgeführt wird. Der Delta-Sigma-Konverter **204a** könnte der Delta-Sigma-Konverter sein, welcher in der Signalfußkurve von **Fig. 11** gezeigt ist, welcher eine Verbesserung der Delta-Sigma-Konverter zweiter Ordnung ist, welche in **Fig. 8** und **Fig. 9** gezeigt sind. Der Demodulator von **Fig. 10** wurde durch ein C Programm simuliert, welches in **Fig. 17** gezeigt. **Fig. 13** bis **Fig. 16** zeigen die Leistung dieses Schaltkreises, wobei der Korrekturblock für zwei unterschiedliche Signalpegel ein- und ausgeschaltet wird bzw. wurde.

[0071] Dasselbe Numerierungsschema ist in **Fig. 10** verwendet, wie es in **Fig. 5** verwendet wurde, wo die

Blöcke gleichwertig sind. Der Delta-Sigma-Modulator **204a** von [Fig. 10](#) ist leicht unterschiedlich vom Delta-Sigma-Modulator **204** von [Fig. 5](#), wie unten beschrieben wird. Hochauflösende Daten **202** werden in den Delta-Sigma-Modulator **204a** eingegeben. Für das Beispiel von [Fig. 10](#) betragen hochauflösende Daten **202** sechzehn Bit. Der Delta-Sigma-Modulator **204a** gibt Daten **206** mittlerer Auflösung, in diesem Fall fünf Bit von Daten entsprechend 17 Pegeln, an den Lastverhältnis-Demodulator **208** aus. Ein Unterschied zwischen dem Demodulator von [Fig. 5](#) und jenem von [Fig. 10](#) ist, daß der Hochtakt **212** durch den Delta-Sigma-Modulator **204a** von [Fig. 10](#) verwendet wird, um den Drei-Stufen-Addierer **234** zu implementieren, wie unten beschrieben wird.

[0072] Der Schaltkreis von [Fig. 10](#) ist sehr effizient, da dieser einen Drei-Stufen-Addierer **234** verwendet, um alle drei der Eingangsaddierer (**241**, **244** und **250**) von [Fig. 11](#) zu implementieren. Dies ist möglich, da die Addierer immer in Folge arbeiten. Der Ausgang des Addierers **241** in [Fig. 11](#) speist in den Addierer **244** ein, welcher in den Addierer **248** einspeist. Jeder Addiervorgang wird durch den Addierer **234** durchgeführt, während die Eingänge zum Addierer **234** durch den Logikblock **232** und Multiplexer **227** bestimmt werden. Der Eingang zum Register **236** ist gesperrt, wenn der Addierer **234** als einer der ersten beiden Addierer (**241**, **244**) agiert, und ist freigegeben, wenn der Addierer **234** als der dritte Addierer **248** wirkt.

[0073] Somit schreitet der Delta-Sigma-Modulator **204a** von [Fig. 10](#) durch die drei Addiererstufen folgendermaßen. Addiererstufe 0 (**241** in [Fig. 11](#)) hat als ihre Eingänge: hoch auflösende Eingangsdaten **202** ausgewählt durch MUX **227** und weiter geleitet zu Addierer **234** (**240** in [Fig. 11](#)), den verzögerten Ausgang des ersten Addierersignals **235**, weitergeleitet durch Register **228** (Signal **258** ist durch die Verzögerung **259** durchgegeben, um das verzögerte Signal **261** in [Fig. 11](#) zu bilden), und Rückkopplungssignal **238**, welches ausgewählt und weitergeleitet zum Addierer **234** durch die Logik **232** (Rückkopplungssignal **254** in [Fig. 11](#)) ist.

[0074] Addiererstufe 1 (**244** in [Fig. 11](#)) hat als ihre Eingänge: den Ausgang der ersten Addiererstufe, gespeichert im Register **228** und bereitgestellt zum Addierer **234** (Signal **243** in [Fig. 11](#)), einen Korrekturfaktor (Korrekturblock **255** bildet das korrigierte Feedback- bzw. Rückkopplungssignal **256** in [Fig. 11](#)), und den verzögerten Ausgang der zweiten Addiererstufe, welche im Register **230** gespeichert und ausgewählt durch MUX **227** ist (Addierer **244** Ausgang **247** ist verzögert durch Verzögerung **264**, um das verzögerte Signal **265** in [Fig. 11](#) zu bilden). Sowohl für Addiererstufe 0 als auch Addiererstufe 1 blockiert das Register **236** das Signal **235**.

[0075] Addiererstufe 2 (**248** in [Fig. 11](#)) hat als ihre Eingänge: ein Zitter- bzw. Rauschsignal, um das System am Erzeugen von Tönen zu hindern, gebildet durch die Logik **232** und weitergereicht zum Addierer **234** (erzeugt durch Block **267** in [Fig. 11](#)), den Ausgang der ersten Addiererstufe, noch im Register **228** gespeichert und dem Addierer **234** (Signal **245** in [Fig. 11](#)) bereitgestellt, und den Ausgang der zweiten Addiererstufe **229**, befördert durch das Register **230** zu MUX **227** (Signal **247** in [Fig. 11](#)). Da die Resultate dieser Addiererstufe **2** ausgegeben werden (nachdem sie durch den Quantisierer **251** und die eingebaute Verzögerung **253** in [Fig. 11](#) weiter gereicht werden), akzeptiert das Register **236** das Sechzehn-Bit-Ausgangssignal **235** vom Addierer **238** und quantisiert es, um es als Daten **206** mittlerer Auflösung (5 Bit) zum Lastverhältnis-Demodulator **208** auszugeben.

[0076] Der Takt- und Zeitgeberblock **239** stellt den mittleren Takt **213**, 0,5 mittleren Takt **214** (wenn verwendet) und hohen Takt **212** zur Verfügung. In [Fig. 5](#) werden nur der mittlere Takt **213** (und 0,5 mittlere Takt **214**, wenn verwendet) durch den Delta-Sigma-Modulator **204** benötigt, da jeder Addierer getrennt implementiert ist, und kein Bedarf zum Arbeiten auf einer höheren Frequenz als der mittlere Takt besteht. Für den Delta-Sigma-Modulator **204a** von [Fig. 10](#) werden jedoch von dem hohen Takt **212** abgeleitete Signale durch den Multiplexer (MUX) **227**, das Register **228** und **230** und die Logik **232** benötigt, um drei Stufen eines Addierens in die für einen Rahmen von Ausgangsdaten erlaubte Zeitlinie einzupassen. Das Register **236** erfordert nur den mittleren Takt **213**. Wenn 0,5 mittlerer Takt **214** verwendet wird (da das durch den Lastverhältnis-Demodulator **208** angewendete Format abwechselt, was erfordert, daß eine Korrekturlogik innerhalb des Logikblocks **232** abwechselt) wird der 0,5 mittlere Takt **214** im Logikblock **232** und dem Lastverhältnis-Demodulator **208** zur Verfügung gestellt.

[0077] Offensichtlich läuft der hohe Takt **212** auf einer höheren Rate bzw. Frequenz, als es erforderlich ist, um drei Addierstufen aufzuweisen. Bis zu sechzehn addierende bzw. Addierstufen könnten innerhalb des Delta-Sigma-Modulators **204a**, falls erforderlich arbeiten, beispielsweise durch einen Delta-Sigma-Modulator höherer Ordnung. Die zusätzlichen Addierstufen können auch verwendet werden, um eine multiple bzw. vielfache Eingangsverarbeitung zu implementieren, wie beispielsweise einen Stereoeingang. In dieser Ausführungsform würde das Eingangssignal **202** den rechten Kanal (beispielsweise) für drei Addierphasen zur Verfügung stellen und würde dann den linken Kanal für drei Phasen zur Verfügung stellen. Alle Register würden zwei Kopien aufweisen. Das Signal **206** mittlerer Auflösung würde zwischen Signalen des linken und rechten Kanals abwech-

seln und eine Schaltanordnung würde bereitgestellt, um den rechten Kanal von dem linken Kanal zu trennen und diese an unterschiedliche Lautsprechern zur Verfügung zu stellen. Irgendwelche Extrazeitphasen werden nicht verwendet.

[0078] [Fig. 12](#) zeigt einen kompensierten Delta-Sigma-Konverter der dritten Ordnung in Übereinstimmung mit der vorliegenden Erfindung. Die Feedback- bzw. Rückkopplungssignale in den zweiten Akkumulator **125** und dritten Akkumulator **130** müssen jedes für das Ausgangsformat des Lastverhältnis-Demodulators **208** (gezeigt in [Fig. 5](#)) kompensiert sein. Diese Kompensationen werden im allgemeinen unterschiedlich für jede Stufe sein.

[0079] Das Eingangssignal x_i , **118** ist bzw. wird mit dem Rückkopplungssignal **135** durch den Addierer **120** kombiniert und in den ersten Akkumulator **121** eingespeist. Der erste Akkumulator **121** umfaßt die Verzögerung **122** und den Addierer **123**. Der Ausgang des Akkumulators **121** ist zu der Rückkopplung **139** und dem Korrekturfaktor **142** vom Korrekturblock **140** hinzuaddiert und in den Akkumulator **125** eingespeist, welcher die Verzögerung **126** und den Addierer **127** umfaßt. Der Ausgang des Akkumulators **125** wird zur Rückkopplung **136** und zu dem Korrekturfaktor **143** vom Korrekturblock **137** hinzu addiert und in den Akkumulator **130** eingespeist, welcher die Verzögerung **131** und den Addierer **132** umfaßt. Der Ausgang des Akkumulators **130** wird durch den Quantisierer **134** quantisiert, um das Ausgangssignal y_i , **119** und das Rückkopplungssignal **135** zu bilden. Korrekturböcke (beispielsweise ROMs) **140** und **137** berechnen Korrekturfaktoren **142** und **143**, wie dies oben beschrieben wurde.

[0080] [Fig. 13](#) bis [Fig. 16](#) zeigt das Ausgangssignal eines Oversampling D/A-Konverters mit und ohne der Korrektur der vorliegenden Erfindung für zwei unterschiedliche Signalpegel bzw. -niveaus. Das D/A-Konvertersystem ist im wesentlichen identisch mit dem D/A-Konverter von [Fig. 4](#), welcher in [Fig. 10](#) und [Fig. 11](#) gezeigte Modulator für den Demodulator **69** verwendet wird. Für [Fig. 13](#) und [Fig. 15](#) ist die Korrektur in [Fig. 10](#) abgeschaltet. Für [Fig. 14](#) und [Fig. 16](#) wird der Korrekturfaktor in [Fig. 10](#) angewendet. Der D/A-Konverter ist in [Fig. 17](#) simuliert, um jedes dieser Ausgangssignale zu erzeugen.

[0081] [Fig. 13](#) zeigt das Spektrum des Ausgangssignals **145**, ohne die Kompensation der vorliegenden Erfindung für ein Eingangssignal, welches aus einer Sinuswelle bei -40 dB besteht. Diese Figur wurde durch die Simulation von [Fig. 17](#) mit abgeschalteter Korrektur erzeugt. [Fig. 14](#) zeigt das Spektrum des Ausgangssignals bei angewendeter Korrektur für ein Eingangssignal, welches eine Sinuswelle bei -40 dB unter Originalgröße umfaßt. Offensichtlich hat sich das Signal/Rausch-Verhältnis bedeutend verbessert.

[0082] [Fig. 15](#) zeigt das Spektrum des Ausgangssignals **145** für ein Eingangssignal, welches eine Sinuswelle bei -60 dB umfaßt. Diese Figur wurde durch die Simulation von [Fig. 17](#) mit abgeschalteter Korrektur erzeugt. [Fig. 16](#) zeigt das Ausgangsspektrum mit Korrektur für ein Eingangssignal, welches eine Sinuswelle bei -60 dB umfaßt. Mit einem derartig niedrigen Eingangssignal wird das Ausgangssignal in dem Rauschen im unkompenzierten Fall nahezu verloren.

[0083] [Fig. 17](#) stellt eine C Programm-Simulation eines Oversampling D/A-Konverters zur Verfügung, welches den kompensierten Delta-Sigma-Quantisierer von [Fig. 10](#) und [Fig. 11](#) verwendet. Diese besondere Ausführungsform wendet ein zentriertes, nach rechts anwachsendes Format, wie das in [Fig. 7A](#) gezeigte an. Das Eingangssignal ist eine 16 Bit, 2 kHz Sinuswelle. Der Ausgang ist ein 1-Bit-Format-Signal, welches die Sinuswelle darstellt, welche durch das System von [Fig. 10](#) ausgegeben würde.

[0084] Geordnet zeigen die Abschnitte des C Programms die Implementierung einer linearen Rückkopplungsfunktion (Teil des Logikblocks **232**), einen Korrekturfaktor, welcher an die Rückkopplung der zweiten Ordnung (Teil des Logikblocks **232**) angewendet und optimiert wird für das zentrierte, nach rechts anwachsende Format, ROM **220** für den Lastverhältnis-Demodulator **208** (zentriertes, nach rechts anwachsendes Format), ein sechzehn Bit-Addierer **234** mit drei Eingängen, welcher sättigt (Überläufe nehmen den Maximalwert und Unterläufe nehmen den Minimalwert an), den Quantisierer **236** (welcher einen Wert im Bereich von 0 bis 16 zurückgibt), Testsignalerzeugung (für Signal **202**), Vorspannungs- (oder Rauschen-) Generator (Teil des Logikblocks **232**), Delta-Sigma-Modulator **204a**, und Lastverhältnis-Demodulator **208**. Arrays bzw. Anordnungen **fb** und **cor** zeigen Feedback- und Korrektursignale, welche für den Lastverhältnis-Modulator geeignet sind, welcher durch den array **out_rom** beschrieben wird. Es ist für Versierte in der Technik selbstverständlich, daß ein Addieren eines Zitter- bzw. Rauschsignals die Qualität des durch das Delta-Sigma-Konvertersystem erzeugten Rauschens verbessern kann, und wird zu diesem Programm hinzugefügt gezeigt.

[0085] Die Implementierung des Delta-Sigma-Modulators **204a** (genannt "main") implementiert die erste Stufe des Addierers **232** (**241** in [Fig. 11](#)), dessen Ausgang "r0" genannt wird. "r1" ist der Ausgang der zweiten Stu-

fe des Addierers **234** (Addierer **244** in [Fig. 11](#)) und "aus" ist die dritte Stufe des Addierers **232** (Addierer **248** in [Fig. 11](#)) nach einer Quantisierung durch den Quantisierer **236** (**251** in [Fig. 11](#)).

[0086] [Fig. 18](#) zeigt eine Hörhilfe, welche ein Mikrofon **300**, ein A/D-Konversionssystem **302**, eine digitale Signalverarbeitung (DSP) **304**, ein D/A-Konversionssystem **306** und einen Lautsprecher **308** umfaßt. Die Komponenten der Hörhilfe von [Fig. 18](#) sind konventionell und gut verstanden, ausgenommen, daß das D/A-Konversionssystem **306** in Übereinstimmung mit der vorliegenden Erfindung modifiziert wurde. In der bevorzugten Ausführungsform ist das D/A-Umwandlungs- bzw. -Konversionssystem **306** ein Oversampling D/A-Konversionssystem, wie das in [Fig. 4](#) gezeigte, wo der Demodulator **69** durch den Demodulator von [Fig. 5](#) ersetzt wurde. Der Delta-Sigma-Modulator **204** kann jeder der in dieser Anmeldung beschriebenen Delta-Sigma-Modulatoren sein, ist jedoch vorzugsweise einer der korrigierten Rückkopplungs-Delta-Sigma-Modulatoren, welche in [Fig. 8](#) – [Fig. 12](#) gezeigt werden. Der Lastverhältnis-Demodulator **208** ist vorzugsweise wie in [Fig. 6](#) gezeigt.

[0087] [Fig. 19](#) zeigt einen Demodulator, welcher einen Delta-Sigma-Modulator **340** der zweiten Ordnung und einen Ausgangs DAC **346** umfaßt. Der Delta-Sigma-Quantisierer **340** der zweiten Ordnung ist sehr ähnlich zu dem in [Fig. 3](#) gezeigten Delta-Sigma-Modulator des Stands der Technik, mit der Ausnahme, daß die Rückkopplung zu jedem Akkumulator für nicht lineare Abweichungen in den Ausgangsdaten kompensiert ist bzw. wird. Wenn es Zeiteffekte gibt, wie beispielsweise eine Mustererzeugung oder Anstiegsgeschwindigkeits- bzw. Flankensteilheitsprobleme, müssen der Korrekturblock **342** und Korrekturblock **344** nicht triviale unterschiedliche Funktionen implementieren, um Abweichungen der ersten und zweiten Ordnung zu berücksichtigen. Beispielsweise kann eine Schleife korrigiert sein, jedoch nicht die andere. Oder jede Schleife kann auf unterschiedliche Weise korrigiert sein. Ein nicht trivialer Unterschied ist, wo eine Rückkopplungsfunktion nicht eine einfache lineare Funktion der anderen ist.

[0088] Der Delta-Sigma-Modulator **340** kann einen Ein-Bit oder Multilevel-Ausgang aufweisen. In dem Fall eines Multilevel-Delta-Sigma-Modulators **340** werden ein Multilevel- bzw. Mehrfachpegel-Quantisierer **50** und ein passender Multilevel DAC **346** verwendet. Der Multilevel DRC **343** in der Rückkopplung ist ebenfalls notwendig. Wenn der Delta-Sigma-Modulator **340** ein Bit ist, der Quantisierer **50** zwei Pegel ist, ist DAC **346** bemerkenswert vereinfacht.

[0089] Beispielsweise in dem Fall, wo der DAC **346** (in diesem Fall ein Bit) in Ausgangsdatenpulsen resultiert, welche jenen in [Fig. 20](#) aufgrund von Anstiegsratenbegrenzungen ähneln, berücksichtigt der Korrekturblock **344** die Unterschiede in der Fläche zwischen dem ersten Puls **314** und dem zweiten Puls **316**. Die Fläche des ersten Pulses **314** ist (beispielsweise) 0,9 und die Fläche des zweiten Pulses **316** ist 1,0, so daß der Korrekturblock **344** das Integral oder die Fläche als eine Rückkopplung bereitstellt. Es ist im allgemeinen undurchführbar, zwei angrenzende Pulse zu erzeugen, welche identische Flächen aufweisen, da der erste Puls eine Anstiegszeit aufweist, während der zweite Puls augenblicklich auf seinem Maximalwert ist. Zusätzlich zum Aufweisen unterschiedlicher Flächen (erstes Moment) weisen die Pulse **314**, **316** unterschiedliche Schwerpunkte (zweites Moment) auf.

[0090] Der Korrekturblock **342** berücksichtigt die Tatsache, daß die Schwerpunkte der Pulse **314** und **316** auch nicht wie vorhergesagt sind, und erzeugt ein Signal basierend auf dem zweiten Integral des Ausgangs. Es ist anzumerken, daß diese beiden Funktionen nicht lineare Kombinationen voneinander sind. Dieses Schema kann auch auf Modulatoren höherer Ordnung ausgedehnt werden, wo jeder Modulator (potentiell) unterschiedliche Rückkopplung empfängt. Es kann für einige Anwendungen als adäquat empfunden werden, beispielsweise einen Modulator der vierten Ordnung zu verwenden und nur den zweiten Integrator zu korrigieren.

[0091] [Fig. 21](#) zeigt einen Puls **322** der Höhe zwei, welche illustriert, daß der Puls nicht die doppelte Fläche eines einzelnen Pulses **324** aufweist. Für den Fall, wo der DAC **346** (in diesem Fall Multilevel) in Ausgangsdatenpulsen resultiert, welche jenen in [Fig. 21](#) ähneln, berücksichtigt der Korrekturblock **344** die Unterschiede in der Fläche zwischen dem unteren bzw. Bodenpuls **324** und dem oberen bzw. Spitzenpuls **326**. Die obere Hälfte **326** des Pulses **322** weist eine geringere Fläche als die untere Hälfte **324** auf. Erneut ist nicht nur die Fläche unterschiedlich zu der erwarteten Fläche, sondern ebenso ist der Schwerpunkt unterschiedlich. [Fig. 19](#) zeigt ein System zum Korrigieren dieser beiden Unvollkommenheiten in den Ausgangsdaten. Die Fläche des unteren Pulses **324** ist (beispielsweise) 0,9 und die Fläche des oberen Pulses **326** ist 0,8, sondern daß der Korrekturblock **344** 9 für einen "1"-Pegel rückkoppeln würde, und 1,7 für einen "2"-Pegel. Der Korrekturblock **342** berücksichtigt die Tatsache, daß der Schwerpunkt des kombinierten Pulses **322** nicht so wie vorhergesagt ist und würde eine Rückkopplung, basierend auf dem zweiten Integralen erzeugen.

[0092] [Fig. 22](#) zeigt eine Serie **332** von Ausgangspulsen **334**, **336**, **338**, welche unterschiedliche Anstiegszei-

ten und Flächen aufweisen. Diese Figur ist eine realistischere Darstellung einer Serie von Pulsen als [Fig. 20](#). Die Anstiegszeit dieser Pulse ist langsam und nicht linear, so daß nicht nur der erste Puls unterschiedlich von dem zweiten Puls sowohl in Fläche als auch Schwerpunkt ist, sondern der zweite Puls unterschiedlich ist von dem dritten.

[0093] Für den Fall, wo der DAC **346** (in diesem Fall ein Bit) in Ausgangsdatenpulsen resultiert, welche jenen in [Fig. 22](#) ähneln, berücksichtigt der Korrekturblock **344** die Unterschiede in der Fläche zwischen nachfolgenden Pulsen **334**, **336**, **338**. Die Fläche des ersten Pulses **334** ist (beispielsweise) 0,9, die Fläche des zweiten Pulses **336** ist 0,95, und die Fläche des Pulses **338** ist 0,98, so daß der Korrekturblock **344** einen unterschiedlichen Wert für alle 3 Pulse produziert. Es ist in Fällen, wie sie in [Fig. 22](#) gezeigt werden, für den Korrekturblock **344** erforderlich, die Geschichte der Ausgangsdaten **36** zu verfolgen, um den Korrekturfaktor abhängig von der Anzahl von benachbarten Pulsen einzustellen, welche er gesehen hat. Im Gegensatz dazu ist die Korrektur der Ausgangsdaten wie derjenigen, die in [Fig. 20](#) gezeigt sind, einfacher, da die Korrekturböcke **342** und **344** nur die Veränderungen im Wert überwachen müssen. Erneut würden die Korrekturböcke für Stufen höherer Ordnung die Rückkopplung für nachfolgende höhere Integrale der Ausgangswellenform erzeugen. Der Block **342** würde die Rückkopplungskorrektur für das zweite Integral erzeugen.

[0094] [Fig. 23](#) zeigt eine andere Anwendung, in welcher die nicht lineare Ausgangsverzerrungskorrektur der vorliegenden Erfindung angewendet werden kann. Der hochqualitative Delta-Sigma-Demodulator **340** umfaßt Delta-Sigma-Modulator **352** mit 33 Pegeln, welche mit dem Mustergenerator **354** verbunden sind, welcher 1-Bit DACs **356** mit vier Ausgängen anspeist. Bezugnehmend auf [Fig. 19](#) ist der Delta-Sigma-Modulator **340** multipelgig, mit einem Quantisierer **50** mit 33 Pegeln und beinhaltet weiters einen DAC **343** mit 33 Pegeln in dem Rückkopplungspfad **54**. Die vier DAC-Ausgänge werden summiert und gefiltert in der analogen Domäne im Block **358**.

[0095] Die Korrekturböcke **342** und **344** werden in diesem Zusammenhang verwendet, um die Unterschiede in dem ersten und zweiten Moment der erzeugten Muster zu korrigieren. In diesem Fall ist **344** (im allgemeinen) eine lineare Funktion, und **342** korrigiert die Zeitschwankung, welche durch die erzeugten Muster verursacht wird.

[0096] Der Mustergenerator **354** arbeitet ein wenig wie ein PWM mit vier Ausgängen. Der Delta-Sigma-Modulator arbeitet auf einem mittleren Takt und der Mustergenerator erzeugt Ausgänge bei einer schnellen Taktfrequenz. Die Bitmuster aus dem Mustergenerator **354** sind acht Bit lang, um das Muster mit 33 Pegeln zu implementieren. In diesem Fall beträgt der schnelle Takt **8** mal die Frequenz des mittleren Takts. Die Muster sind wie folgt:

Wert aus dem Modulator **340**:

0
1
2
3
4
5
6

7
8
9
10
...
32

Muster aus dem Mustergenerator **354**:

00000000 00000000 00000000 00000000
00010000 00000000 00000000 00000000
00010000 00001000 00000000 00000000
00010000 00001000 00010000 00000000
00010000 00001000 00010000 00001000
00011000 00001000 00010000 00001000
00011000

00011000 00010000 00001000
00011000 00011000 00011000 00001000
00011000 00011000 00011000 00011000
00111000 00011000 00011000 00011000
00111000 00011100 00011000 00011000

11111111 11111111 11111111 11111111

[0097] Jeder der vier Ausgänge von dem Mustergenerator **354** speist einen 1-Bit DAC **356** (im einfachsten Fall ein Draht). Im allgemeinen werden die vier Ausgänge der DACs **356** summiert durch ein Tiefpaßfilter **358** weitergeleitet, um das endgültige, gereinigte Ausgangssignal zu erzeugen. Jene mit Erfahrung in der Technik werden anerkennen, daß viele andere Muster verwendet werden könnten und viele Pegel (beispielsweise 256 Pegel) implementiert werden könnten. Das Problem des zweiten Moments und die Berechnung der Rückkopplungswerte ist identisch zu jenen, welche in dem Klasse D PWM Beispiel gezeigt werden. Erneut kann diese Technik auf Delta-Sigma-Modulatoren höherer Ordnung ausgedehnt werden.

[0098] **Fig. 24** ist ein Blockdiagramm eines Klasse-D-Leistungsverstärkers **360**, welcher einen Delta-Sigma-Modulator **340** (gezeigt in **Fig. 19**) mit einer Korrekturschaltanordnung verwendet. Direkte digitale Verstärker wie **360** treiben einige zehn bis hunderte von Volt bei bis zu 50 Ampere. Der Delta-Sigma-Modulator **340** speist einen PWM DAC **364**, welcher zwei Verstärker **366** antreibt. Das Tiefpaßfilter **368** reinigt die Signale **370**, **372** von den Verstärkern **366** und stellt ein Signal **376** zum Antreiben des Lautsprechers **374** zur Verfügung.

[0099] **Fig. 25** illustriert, warum die Korrekturschaltanordnung des Delta-Sigma-Modulators **340** notwendig ist. Das Signal **376** ist die tiefgepaßte Differenz der Klasse D Signale **370** und **372**. In der realen Welt ist es sehr schwierig, Anstiegszeiten zu erzielen, welche dieselben sind wie die Abfallzeiten. Daher ist der Signalspuls **388** effektiv später als der Signalpuls **390** (obwohl sie dieselbe Fläche aufweisen). Dies ist ein Anstiegsgeschwindigkeitsproblem ähnlich zu jenem, wie es in **Fig. 20-Fig. 22** gezeigt wird und oben besprochen wurde. Die ersten Momente der beiden Pulse sind in diesem Fall identisch, jedoch die zweiten Momente unterscheiden sich, und diese Fehler werden durch die Korrekturschaltanordnung des Delta-Sigma-Modulators **340** korrigiert.

[0100] Jene mit Erfahrung in der Technik werden anerkennen, daß Delta-Sigma-Quantisierer höherer Ordnung, welche (einen) zusätzliche(n) getrennten Korrekturblock (-blöcke) in der Rückkopplung für den (die) zusätzlichen Akkumulator(en) korrigiert, verwendet werden könnten, um Verzerrungen höherer Ordnung in den Ausgangsdaten (drittes Moment und aufwärts) zu kompensieren. Während durch verschiedene Ausgangskonfigurationen verursachte Verzerrungen als Beispiele von Verzerrungen verwendet wurden, welche durch das vorliegende Schema korrigiert werden können, können beliebige systemische, vorhersehbare Verzerrungen der Ausgangsdaten in ähnlicher Weise berücksichtigt werden.

[0101] Funktionen, welche durch die Korrekturböcke **342**, **344** implementiert wurden, müssen nicht trivial unterschiedlich sein, um nicht lineare Verzerrungen zu kompensieren. Ein Beispiel kann eine Implementierung eines quadratischen Gesetzes sein. Die Korrekturböcke **342**, **344** könnten als Suchtabellen oder Logik implementiert werden, welche mathematische Ausdrücke implementiert.

Patentansprüche

1. Kompensierter Delta-Sigma-Digital/Analog-Wandler (DAC) wenigstens zweiter Ordnung, welcher wenigstens erste und zweite Feedback- bzw. Rückkopplungsschleifen aufweist, welche ein ursprüngliches Feedback- bzw. Rückkopplungssignal tragen, umfassend:

Mittel zum Auswählen einer ersten korrigierten Rückkopplung (**342**), basierend auf der ursprünglichen Rückkopplung (**54**), welche an der ersten Rückkopplungsschleife anzulegen ist;

Mittel zum Anlegen bzw. Anwenden der ersten korrigierten Rückkopplung an die erste Rückkopplungsschleife;

Mittel zum Auswählen einer zweiten korrigierten Rückkopplung (**344**), basierend auf der ursprünglichen Rückkopplung (**54**), welche an die zweite Rückkopplungsschleife anzulegen ist; und

Mittel zum Anlegen bzw. Anwenden der zweiten korrigierten Rückkopplung an die zweite Rückkopplungsschleife,

gekennzeichnet dadurch:

daß er weiters Mittel in den Mitteln zum Auswählen einer zweiten korrigierten Rückkopplung (**344**) beinhaltet, um die Geschichte von Ausgangs- bzw. Ausgabedaten (**36**) zu verfolgen, um die zweite korrigierte Rückkopplung in Abhängigkeit von der Anzahl von benachbarten Pulsen einzustellen, welche sie gesehen hat; und worin:

die erste korrigierte Rückkopplung nicht eine lineare Funktion der zweiten korrigierten Rückkopplung ist; und die erste und zweite korrigierte Rückkopplung, wenn sie an die entsprechenden Rückkopplungsschleifen angelegt sind, dazu dienen, Unvollkommenheiten bzw. Fehler (**312**, **332**) in der Ausgabe des DAC zu korrigieren, welche durch Schwankungs- bzw. Anstiegsratenbeschränkungen bzw. Anstiegsgeschwindigkeitsbeschränkungen bewirkt sind.

2. Kompensierter Delta-Sigma-DAC nach Anspruch 1, wobei die erste korrigierte Rückkopplung eine lineare Funktion des ursprünglichen Rückkopplungssignals ist und die zweite korrigierte Rückkopplung eine nicht-lineare Funktion der ursprünglichen Rückkopplung ist.

3. Kompensierter Delta-Sigma-DAC nach Anspruch 1, wobei die erste korrigierte Rückkopplung das ursprüngliche Rückkopplungssignal ist und die zweite korrigierte Rückkopplung eine nicht-lineare Funktion der ursprünglichen Rückkopplung ist.

4. Kompensierter Delta-Sigma-DAC nach einem vorangehenden Anspruch, welcher konfiguriert ist, um ein

Ein-Bit-Ausgangssignal zu erzeugen bzw. zu generieren.

5. Kompensierter Delta-Sigma-DAC nach einem der Ansprüche 1 bis 3, welcher konfiguriert ist, um ein Multilevel-Ausgangssignal zu erzeugen.

6. Ausgabe- bzw. Ausgangsleistungsverstärker der Klasse D, welcher einen kompensierten Delta-Sigma-DAC umfaßt, wie er in Anspruch 4 oder Anspruch 5 beansprucht ist, wobei der Verstärker weiters beinhaltet:

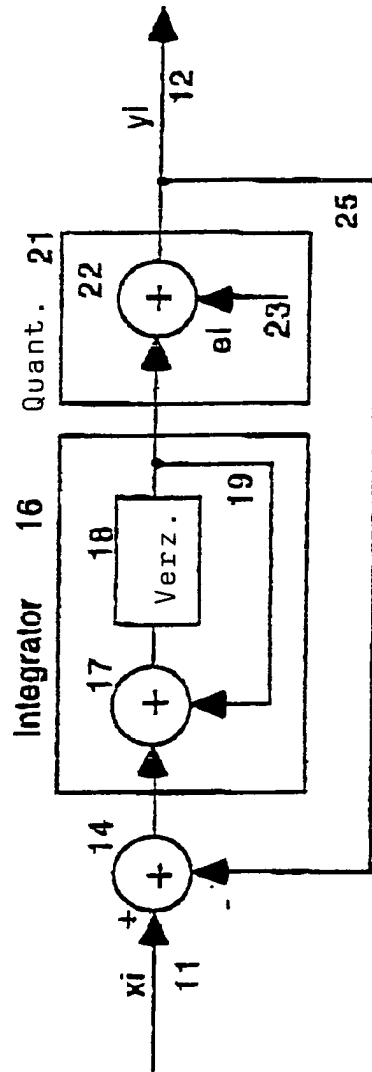
einen Pulswellenmodulator-Digital-Analog-Wandler (**364**), welcher mit dem Ausgang des kompensierten Delta-Sigma-Modulators verbunden ist, zum Erzeugen von zwei analogen bzw. Analogsignalen P und N, basierend auf dem Ausgangssignal von dem kompensierten Delta-Sigma-Modulator;
wobei die Mittel zum Auswählen der ersten korrigierten Rückkopplung und der zweiten korrigierten Rückkopplung Ausgabe- bzw. Ausgangsfehler aufgrund von Verschiedenheiten bzw. Mißverhältnissen in Anstiegs- zu Abfallszeiten in den P- und N-Signalen korrigieren.

7. Verstärker nach Anspruch 6, weiters umfassend:

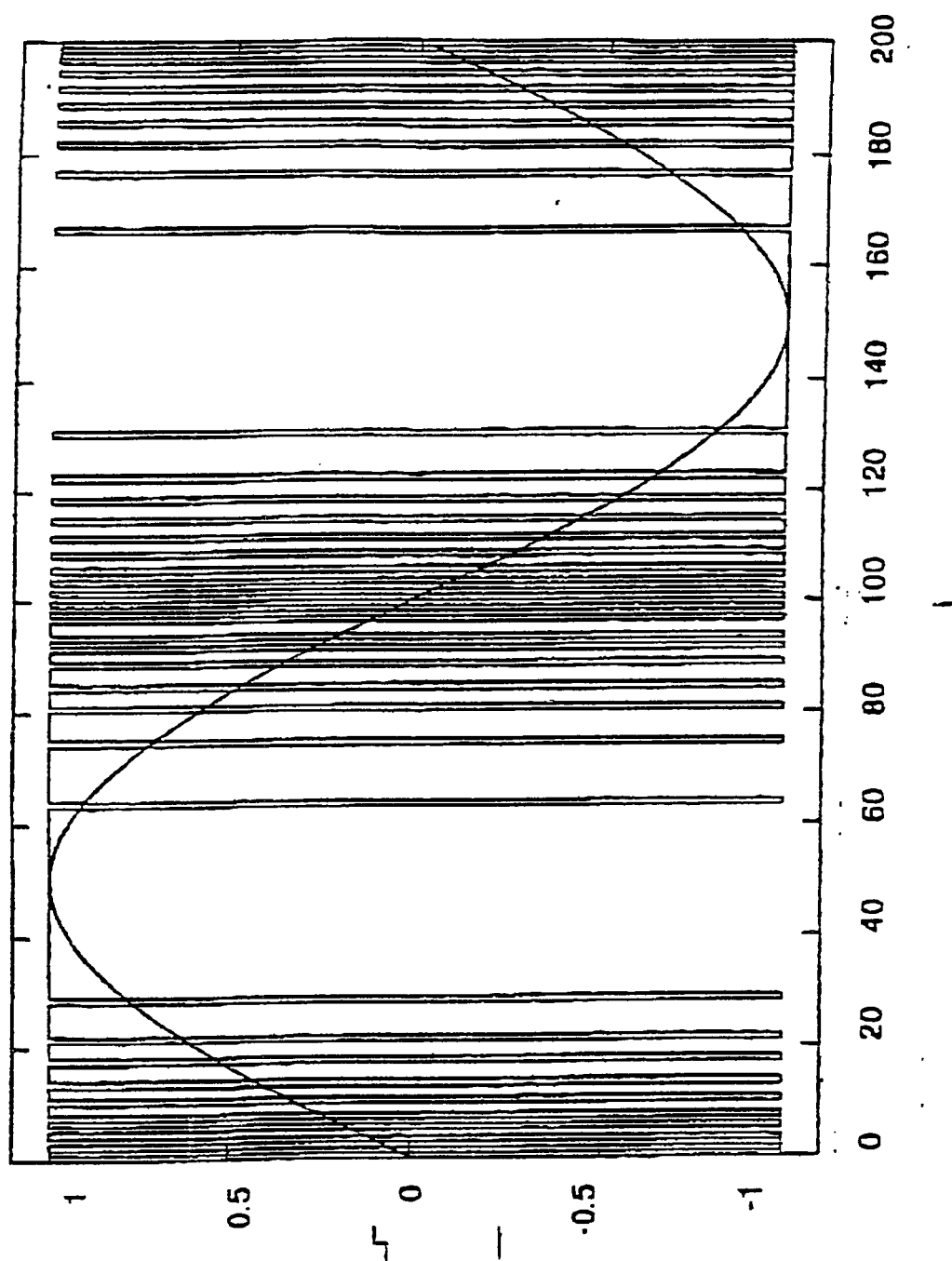
ein Lautsprecherfilter (**368**) zum Erzeugen des gefilterten Signals P-N; und

einen Lautsprecherverstärker (**374**) zum Umwandeln des Signals P-N in ein Audiosignal.

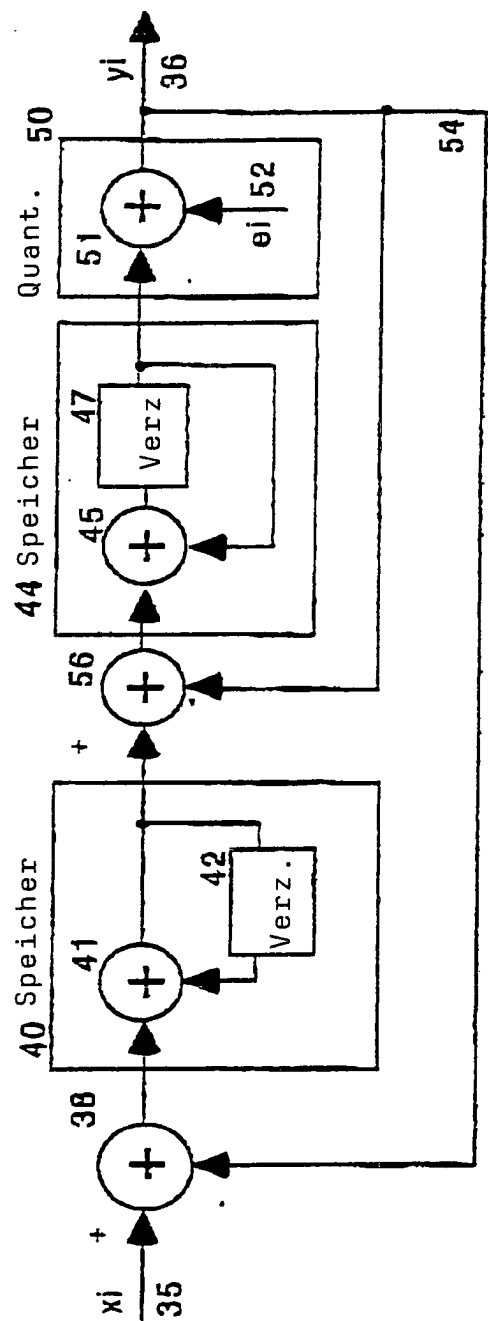
Es folgen 29 Blatt Zeichnungen



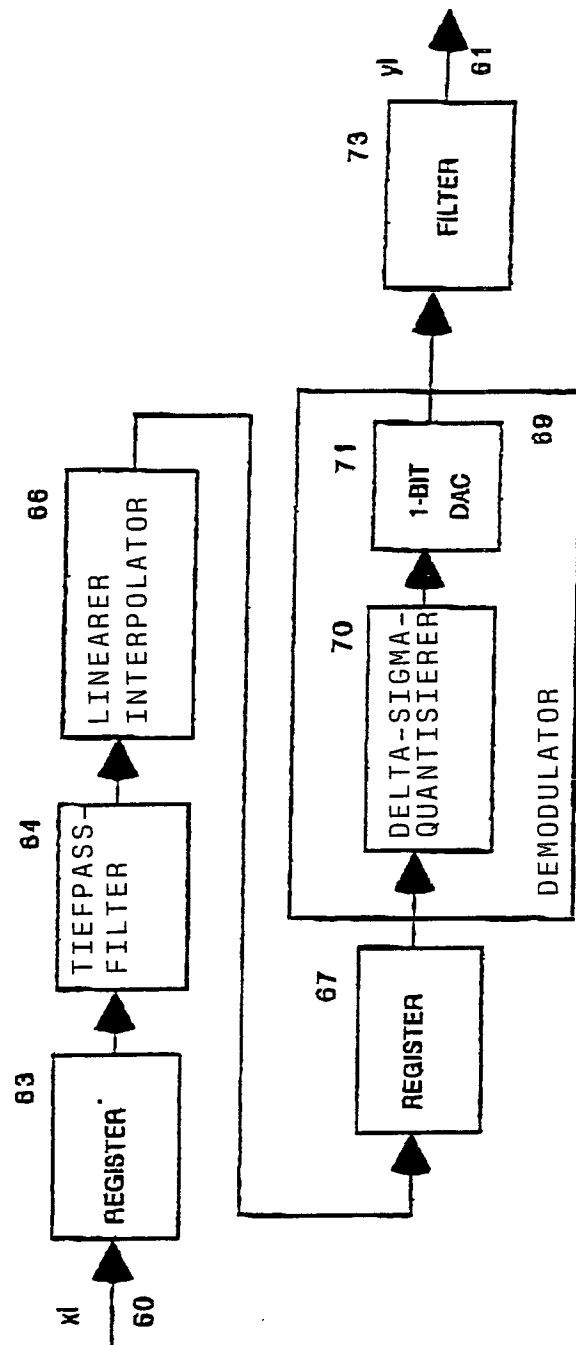
FIGUR 1 STAND DER TECHNIK



FIGUR 2

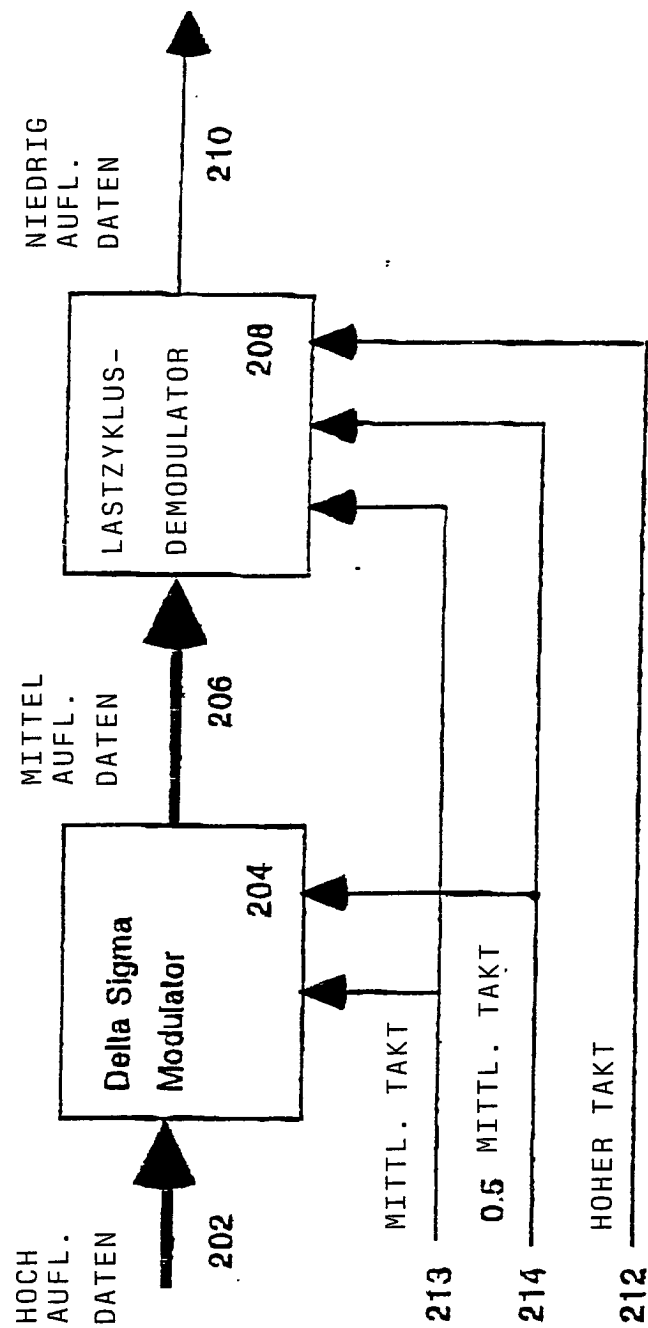


FIGUR 3 STAND DER TECHNIK

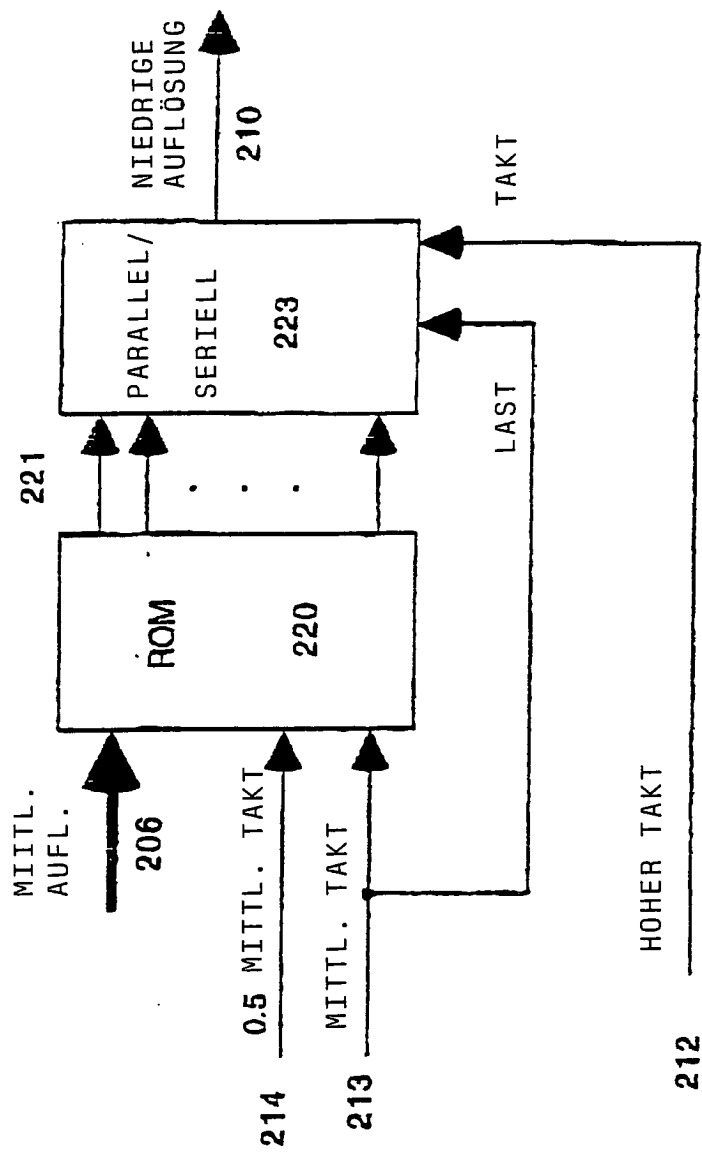


FIGUR 4

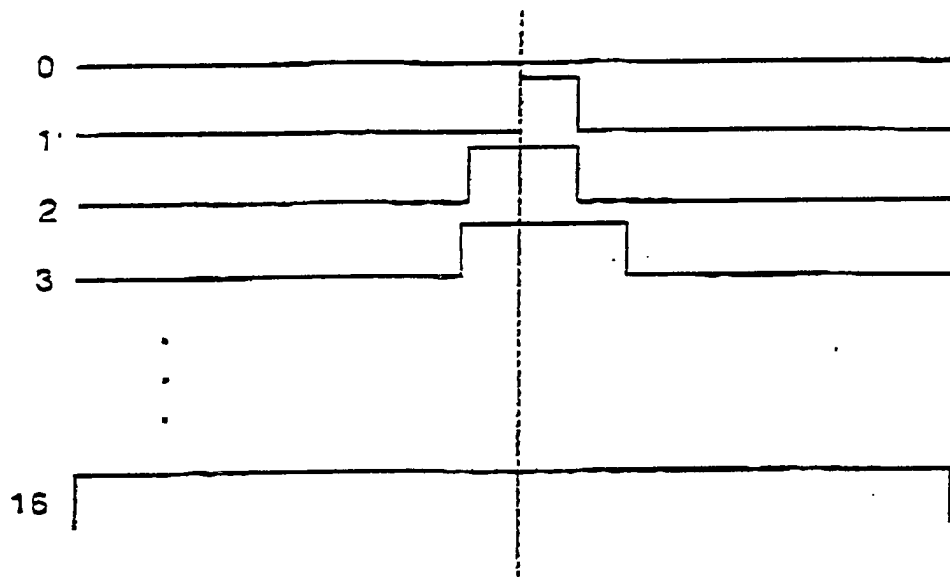
STAND DER TECHNIK



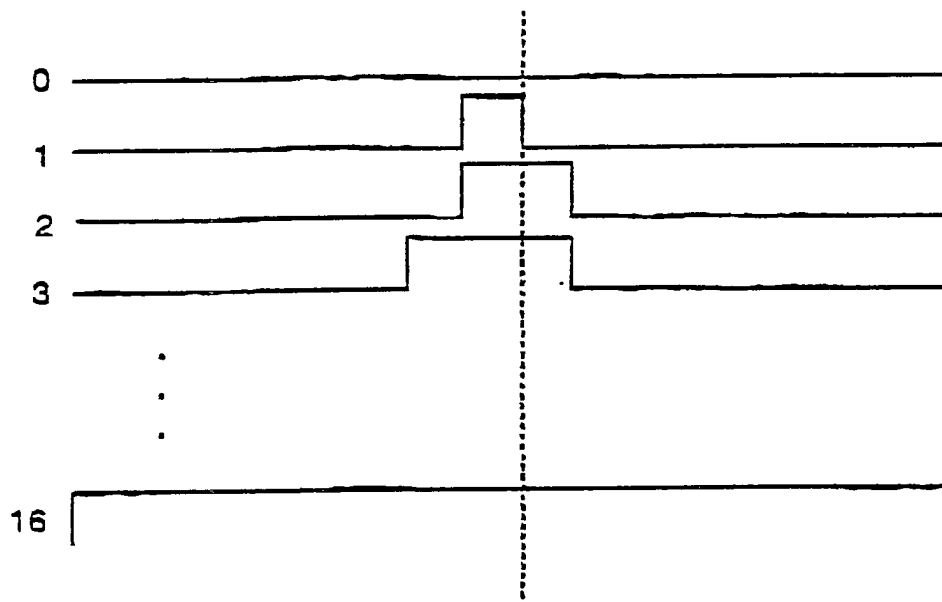
FIGUR 5



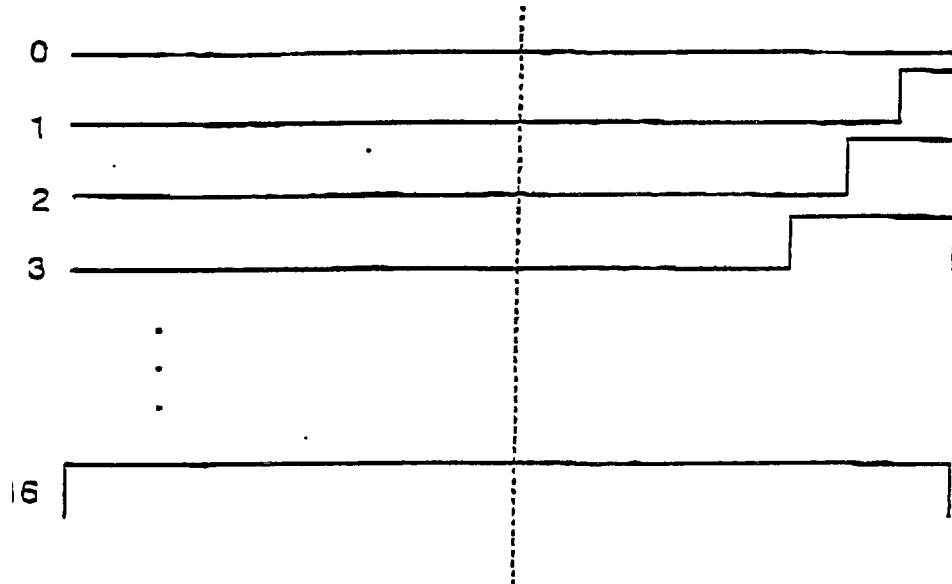
FIGUR 6



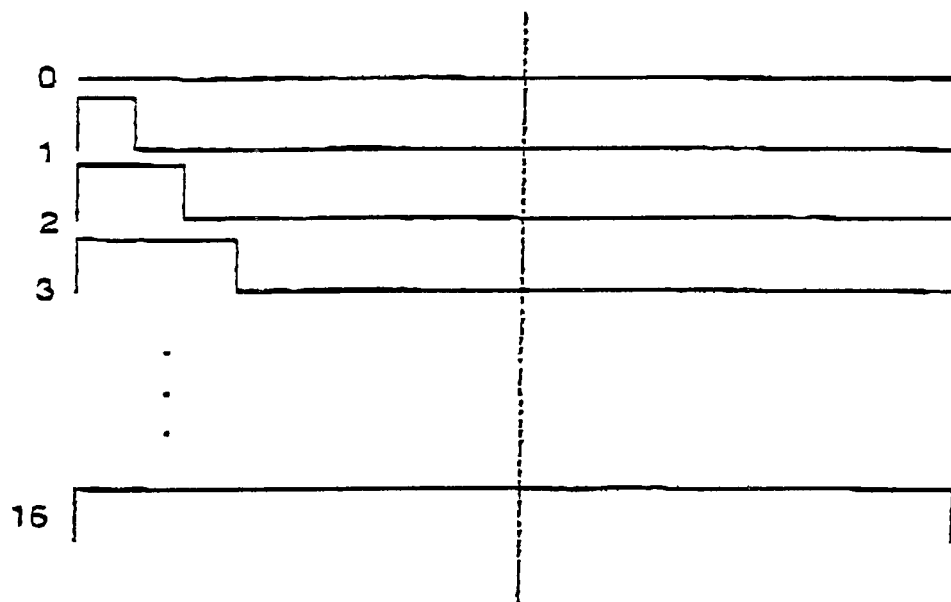
FIGUR 7A



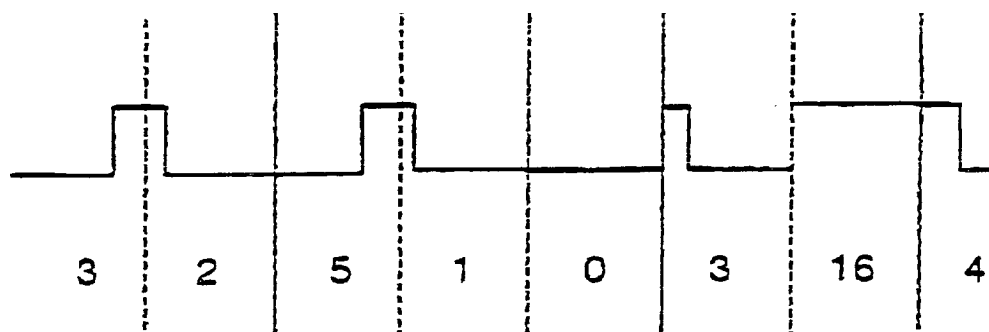
FIGUR 7B



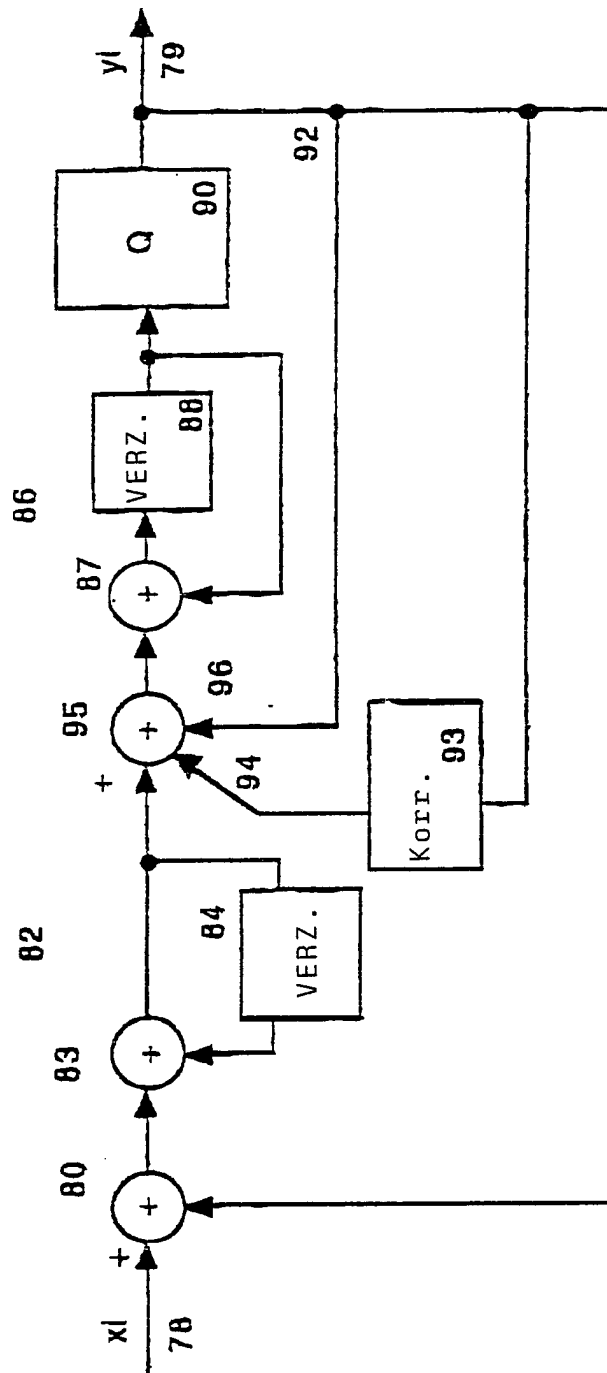
FIGUR 7C



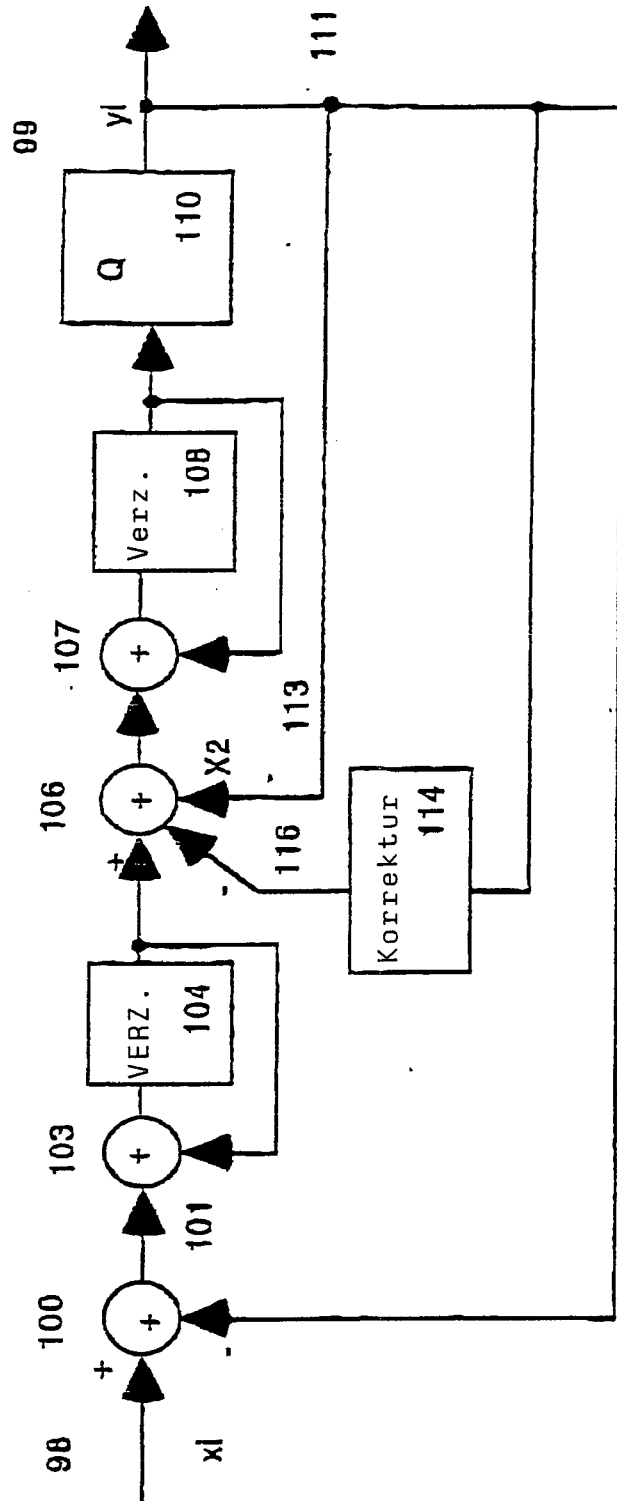
FIGUR 7D



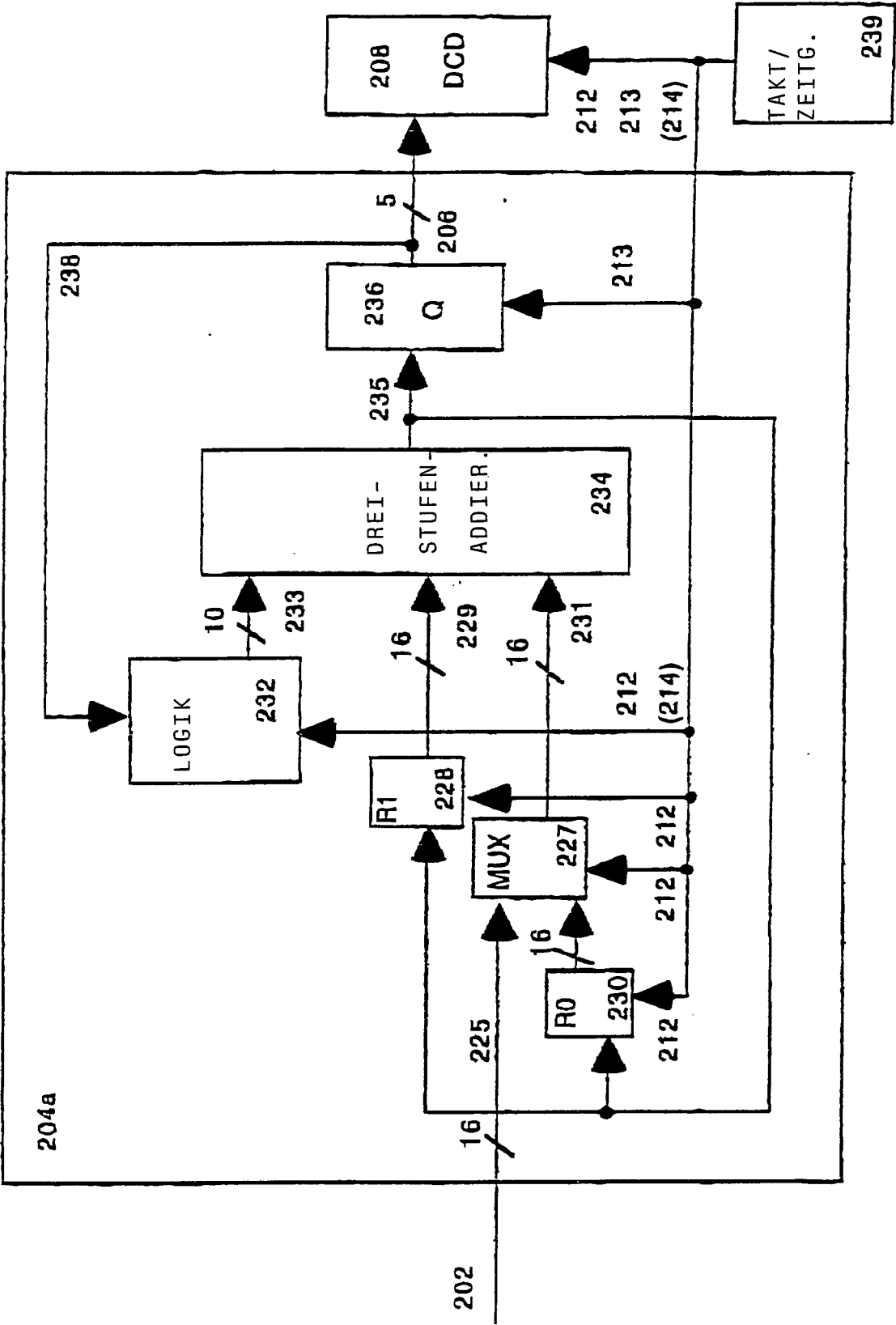
FIGUR 7E



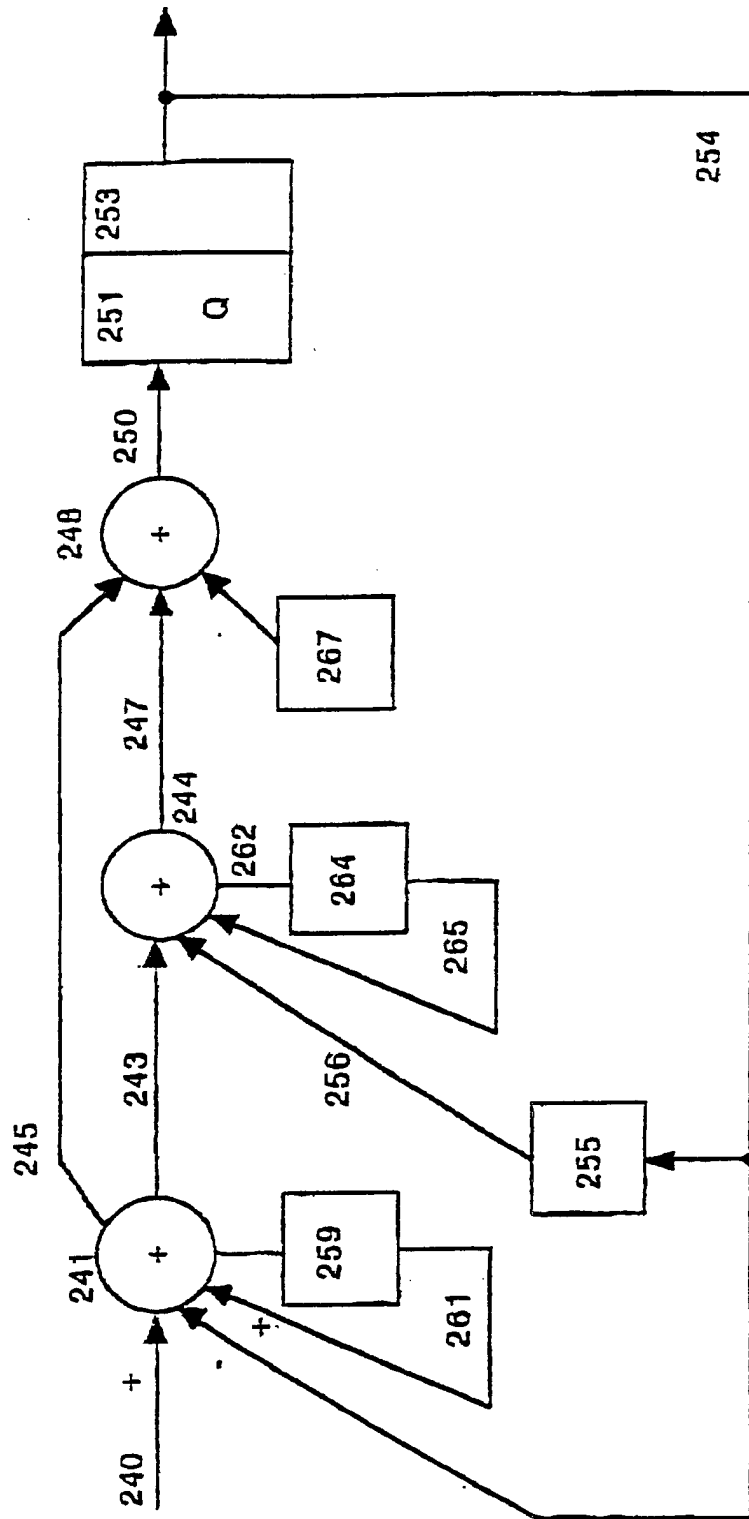
FIGUR 8



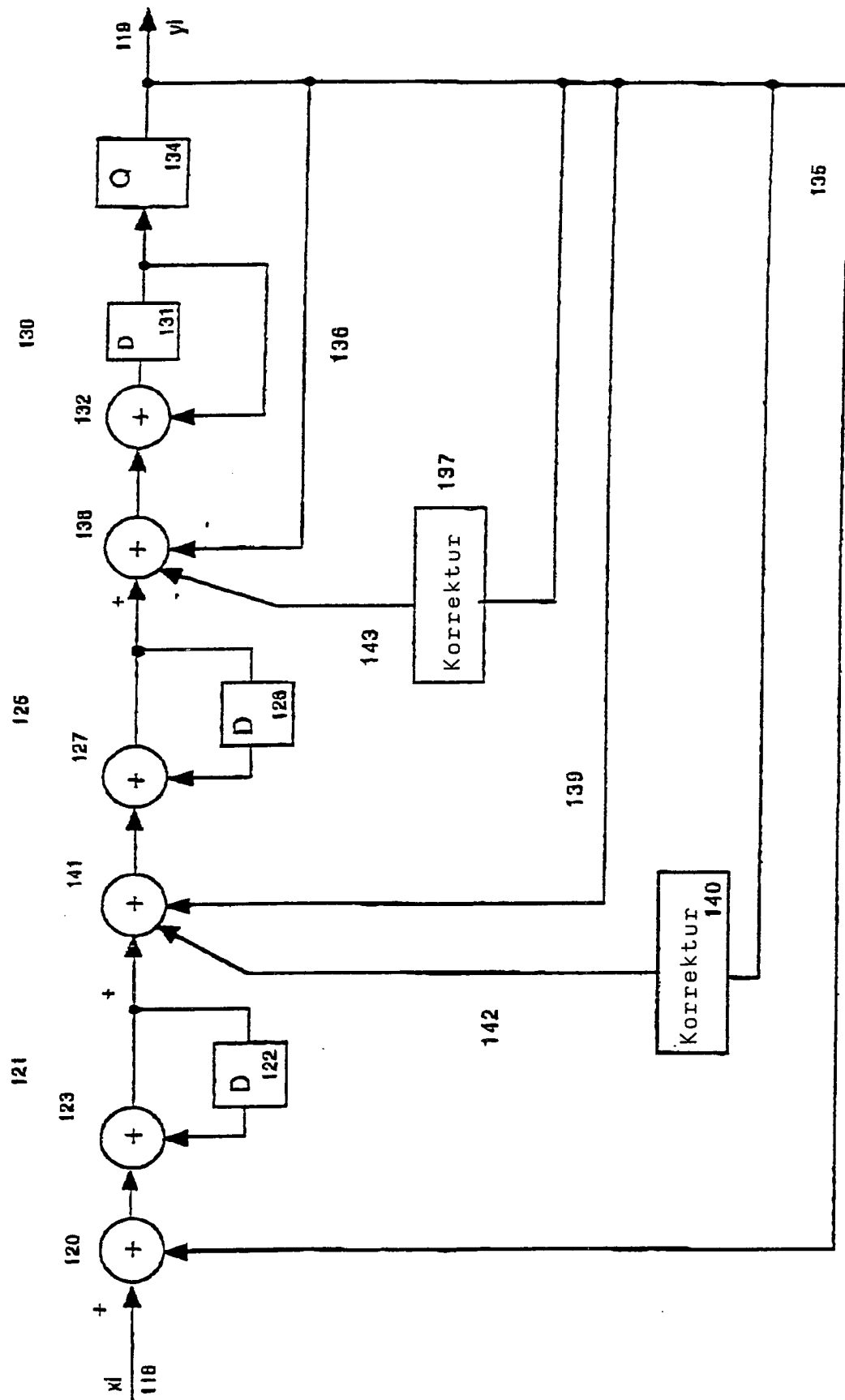
FIGUR 9



FIGUR 10

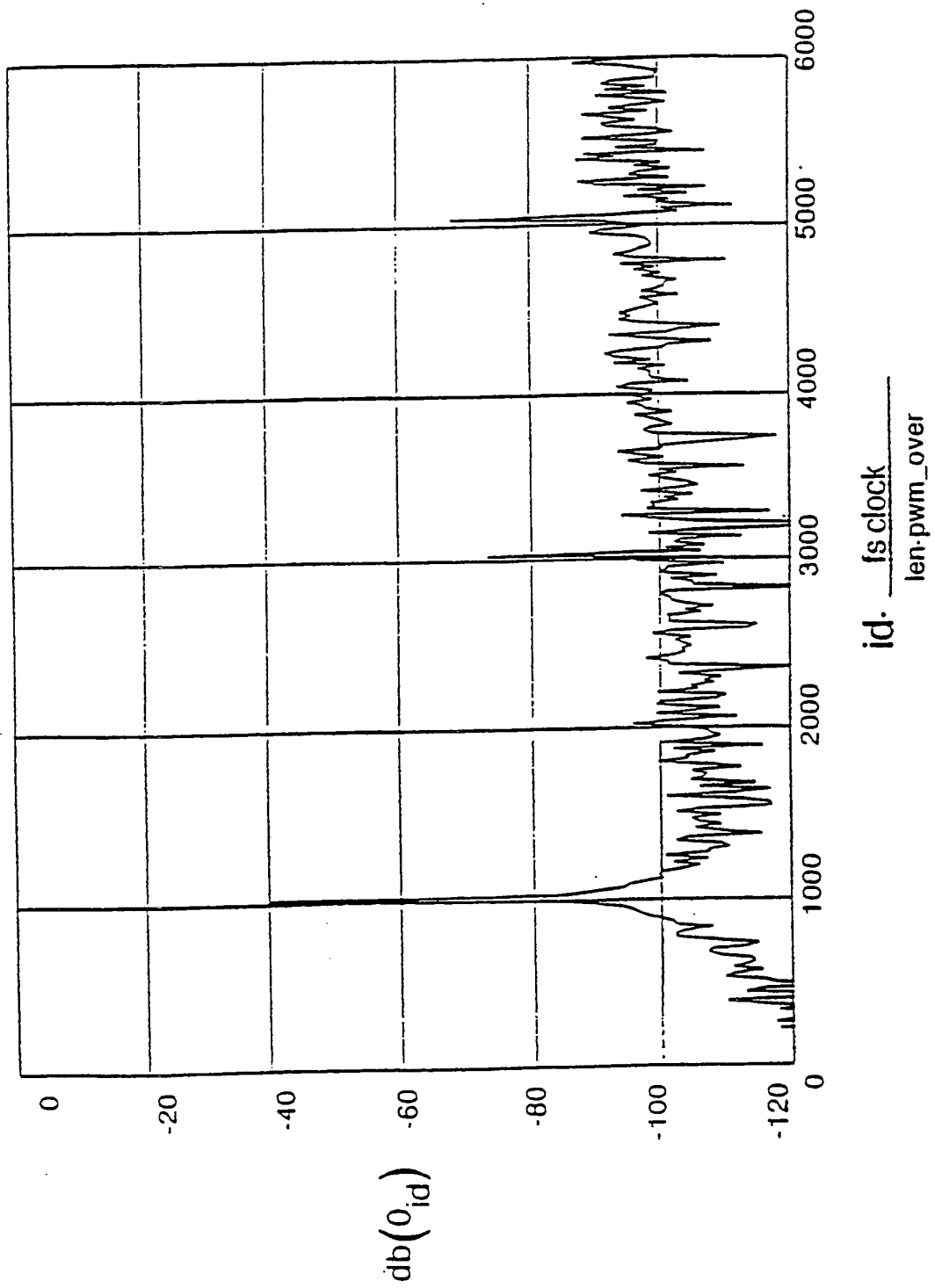


FIGUR 11

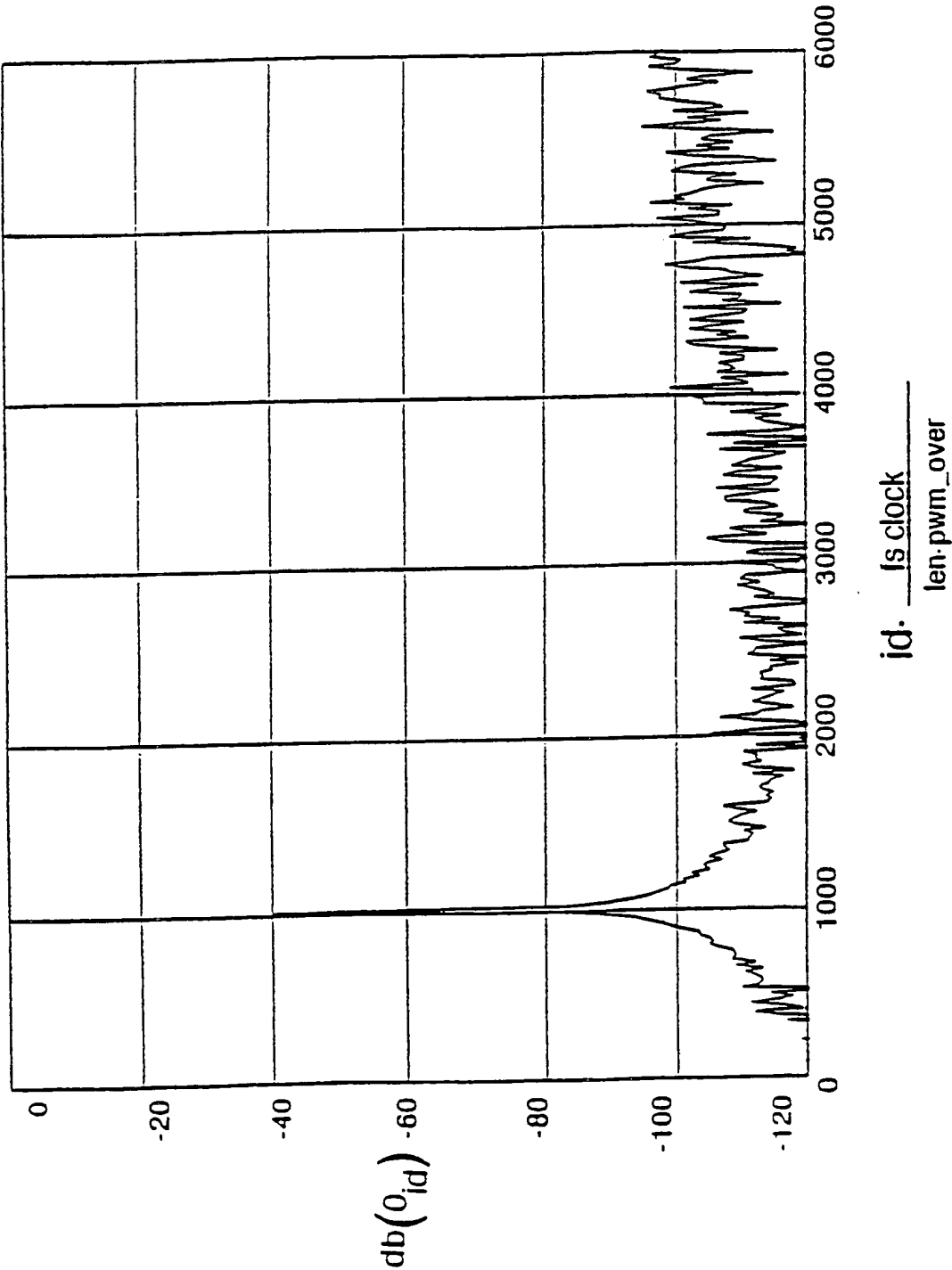


FIGUR 12

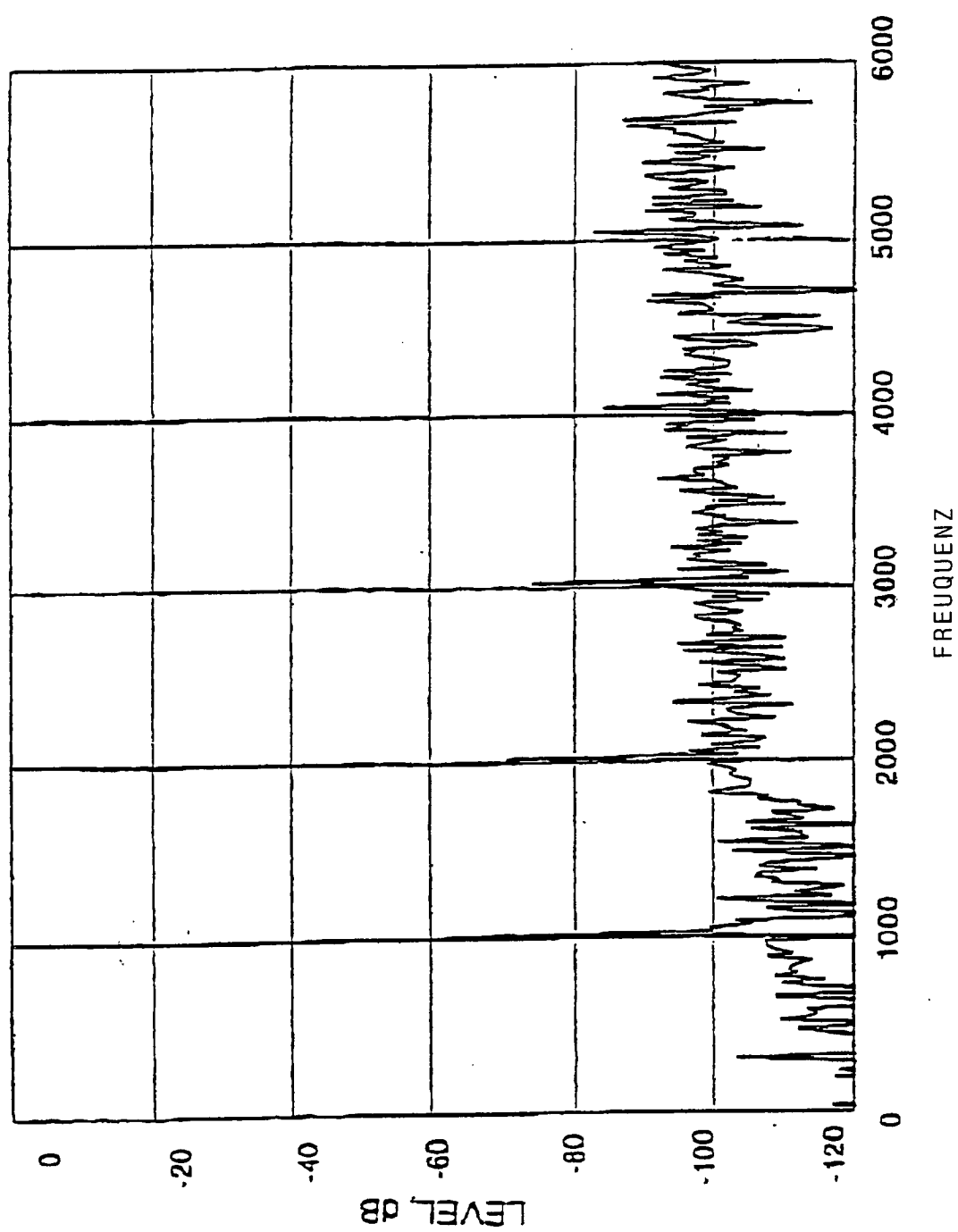
FIGUR 13



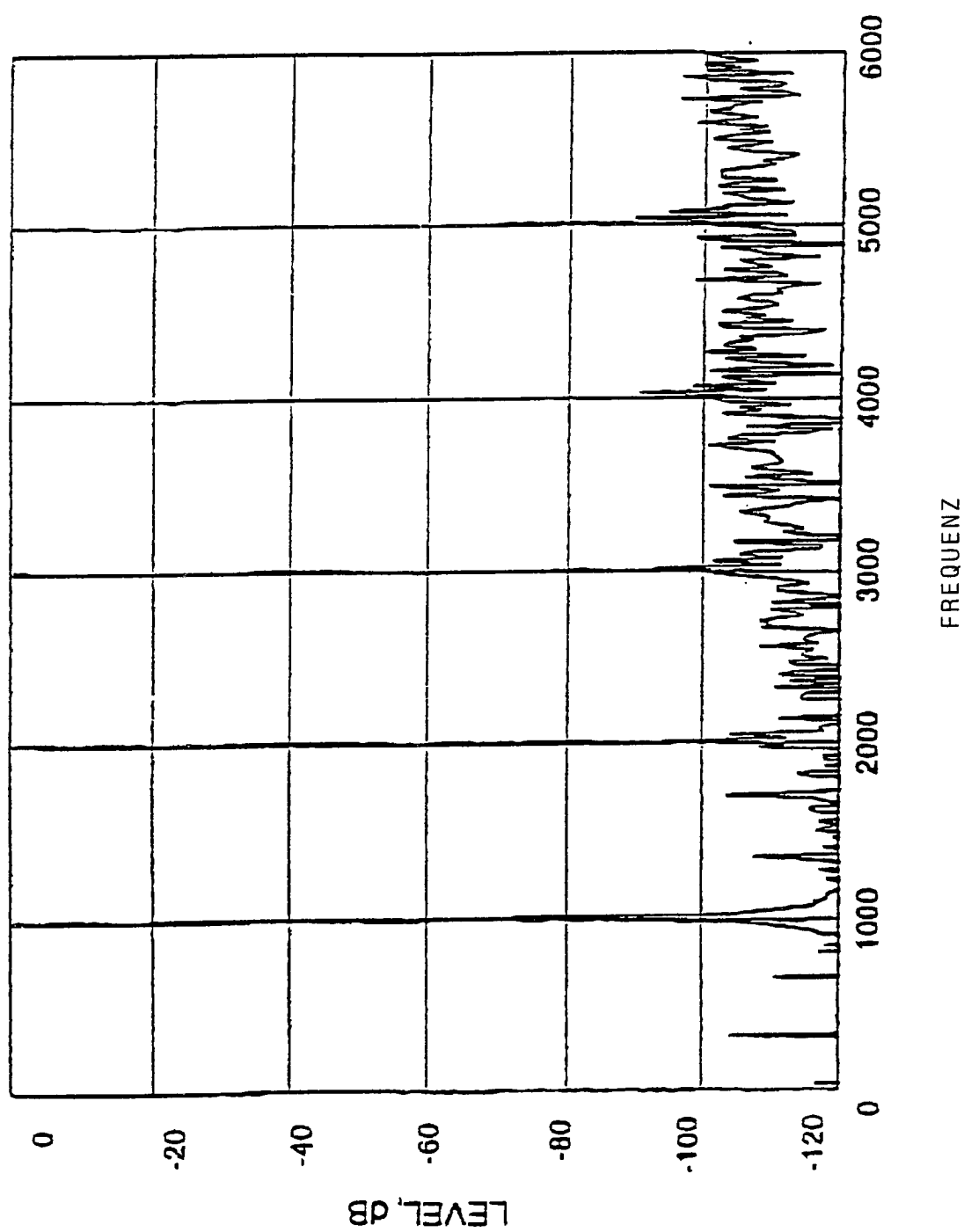
FIGUR' 14



FIGUR . 15



FIGUR 16



```

.....
*
* C program to simulate a noise shaped, PWM digital to analog converter
*
* John Melanson, 20 May 1996
* Assumes an input signal, 16 bit signed integer,
* with +-16*1024 being the input saturation points
*
* The PWM output is operating at 16 times the noise shaping frequency.
*
* Uses a noise shaper than can be built with one 3 port adder
* (two 16 bit inputs, one 10 bit input),
* and operated in three phases.
*
* The delta sigma structure used is especially appropriate,
* as it operates in three phases, and is very
* stingy on hardware resources
*
* The converter is assumed to be operating at a 250 kHz noise shaped rate,
* and at a 4 MHz output bit rate
*
.....
#include <stdio.h>
#include <math.h>

double fs = 250000.;

// Feedback of quantized output

short fb[17] = {
    0x4000, // 0
    0x3800, // 1
    0x3000, // 2
    0x2800, // 3
    0x2000, // 4
    0x1800, // 5
    0x1000, // 6
    0x0800, // 7
    0x0000, // 8
    0xf800, // 9
    0xf000, // 10
    0xe800, // 11
    0xe000, // 12
    0xd800, // 13
    0xd000, // 14
    0xc800, // 15
    0xc000, // 16
};

// Correction factor for second integral. Note that the sign is opposite
// that of the feedback, as odd output widths are late relative to
// even output widths

short cor[17] = {
    0x0000, // 0
    0x0040, // 1
    0x0000, // 2
    0x00c0, // 3
    0x0000, // 4
    0x0140, // 5
    0x0000, // 6
    0x01c0, // 7
    0x0000, // 8
    0x0240, // 9

```

FIGURE 17A

```

0x0000, // 10
0x02c0, // 11
0x0000, // 12
0x0340, // 13
0x0000, // 14
0x03c0, // 15
0x0000 // 16
};

// Rom for duty cycle modulator - pulse
// grows from middle, on right first
//

short out_rom[17][16] = {
    { -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1}, //0
    { -1, -1, -1, -1, -1, -1, -1, -1, +1, -1, -1, -1, -1, -1, -1, -1}, //1
    { -1, -1, -1, -1, -1, -1, -1, +1, +1, -1, -1, -1, -1, -1, -1, -1}, //2
    { -1, -1, -1, -1, -1, -1, -1, +1, +1, +1, -1, -1, -1, -1, -1, -1}, //3
    { -1, -1, -1, -1, -1, -1, -1, +1, +1, +1, +1, -1, -1, -1, -1, -1}, //4
    { -1, -1, -1, -1, -1, -1, -1, +1, +1, +1, +1, +1, -1, -1, -1, -1}, //5
    { -1, -1, -1, -1, -1, -1, +1, +1, +1, +1, +1, +1, -1, -1, -1, -1}, //6
    { -1, -1, -1, -1, -1, +1, +1, +1, +1, +1, +1, +1, +1, -1, -1, -1}, //7
    { -1, -1, -1, -1, +1, +1, +1, +1, +1, +1, +1, +1, +1, -1, -1, -1}, //8
    { -1, -1, -1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, -1, -1, -1}, //9
    { -1, -1, -1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, -1, -1, -1}, //10
    { -1, -1, -1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, -1, -1, -1}, //11
    { -1, -1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, -1, -1, -1}, //12
    { -1, -1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, -1, -1, -1}, //13
    { -1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, -1, -1, -1}, //14
    { -1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, -1, -1, -1}, //15
    { +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, +1, -1, -1, -1} //16
};

/*****
 *
 *          add_3
 *
 * Emulate a 3 input, 16 bit adder (the third input, c,
 * needs to be only 10 bits).
 * The adder saturates on overflow and underflow
 *
 *****/

short add_3(short a, short b, short c){
    long temp = (long)a + long(b) + long(c);
    if (temp > (signed)0x00007fffL) return 0x7fff;
    if (temp < (signed)0xffff8000L) return 0x8000;
    return (short) temp;
}

/*****
 *
 *          quant
 *
 * Quantize a 16 bit value to a number in the range 0 - 16. Over and
 * underflows will produce values of 16 and 0, respectively. A 0 input
 * produces an 8 output, which corresponds to a 50% duty cycle
 *
 *****/

short quant(short x){
    if (x >= (signed short) 0x4000) return 16;

```

FIGURE 17B

```

    if (x <= (signed short) 0xc000) return 0;
    return (x + 0x4000) >> 11;
}

/*****
 *
 *      insig
 *
 *  Make a test signal, 2 kHz, +- 10000 amplitude. This function
 *  would normally be replaced by a signal source
 *  followed by some kind of sample rate increasing system.
 *
 *****/

#define testfreq 2000.
#define level 10000.

short insig( long time){
    return (short)(.5 + level * sin(2 * 3.14159 / fs * testfreq * (long)time));
}

/*****
 *
 *      bias
 *
 *  The output of this routine could be a simple 0x0400, but better
 *  results are had with a little bit of white noise injected.
 *  White noise comes from a random prn
 *
 *****/

short bias(void){
    static short r = 1;
    r = (r << 1) | (1 & ((r>>14)^(r>>13)) );
    //
    // return 0x300,0x340,0x380 ... 0x6c0
    //
    return (r & 0x3c0) + 0x300;
}

/*****
 *
 *      main
 *
 *  Emulate converter
 *  Test will run for 500 us, one cycle of the test sine wave
 *
 *
 *  The three ports of the adder are connected as follows
 *  port      phase 0      phase 1      phase 2
 *  a          r0          r0          r0          (hard wired)
 *  b          input      r1          r1          (requires a 2 input mux)
 *  c          feedback  correction  bias        (requires simple logic)
 *
 *****/

void main(void){
    short r0 = 0; // one of the hardware integrations registers
    short r1 = 0; // the other hardware integration register
    short out = 8; // the previous output duty cycle

```

FIGURE 17C

```

long time;
FILE *results;

/* open an output file for update */
results = fopen("DAC.OUT", "w+");
fprintf(results, "Output results of DAC simulation ~ 4MHz rate\n");

for ( time = 0; time < 125; time ++){

//
// Do delta - sigma part
//

    r0 =          add_3( r0 , insig(time) , fb[out] );    // phase 0
    r1 =          add_3( r0 , r1          , cor[out] );    // phase 1
    out = quant( add_3( r0 , r1          , bias() ) );    // phase 2

//
// Now do duty cycle modulator
//

    for(int tick = 0; tick < 16; tick++){
        fprintf (results, "%+4d", out_rom[out](tick));
    }
    fprintf(results, "\n");
}
fclose(results);
}

```

FIGURE I7D

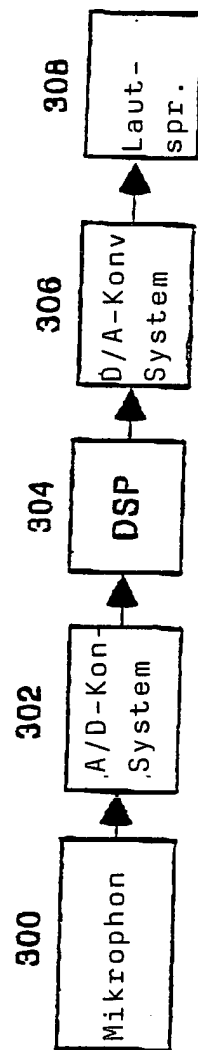
Output results of DAC simulation - 4MHz rate

[illegible]

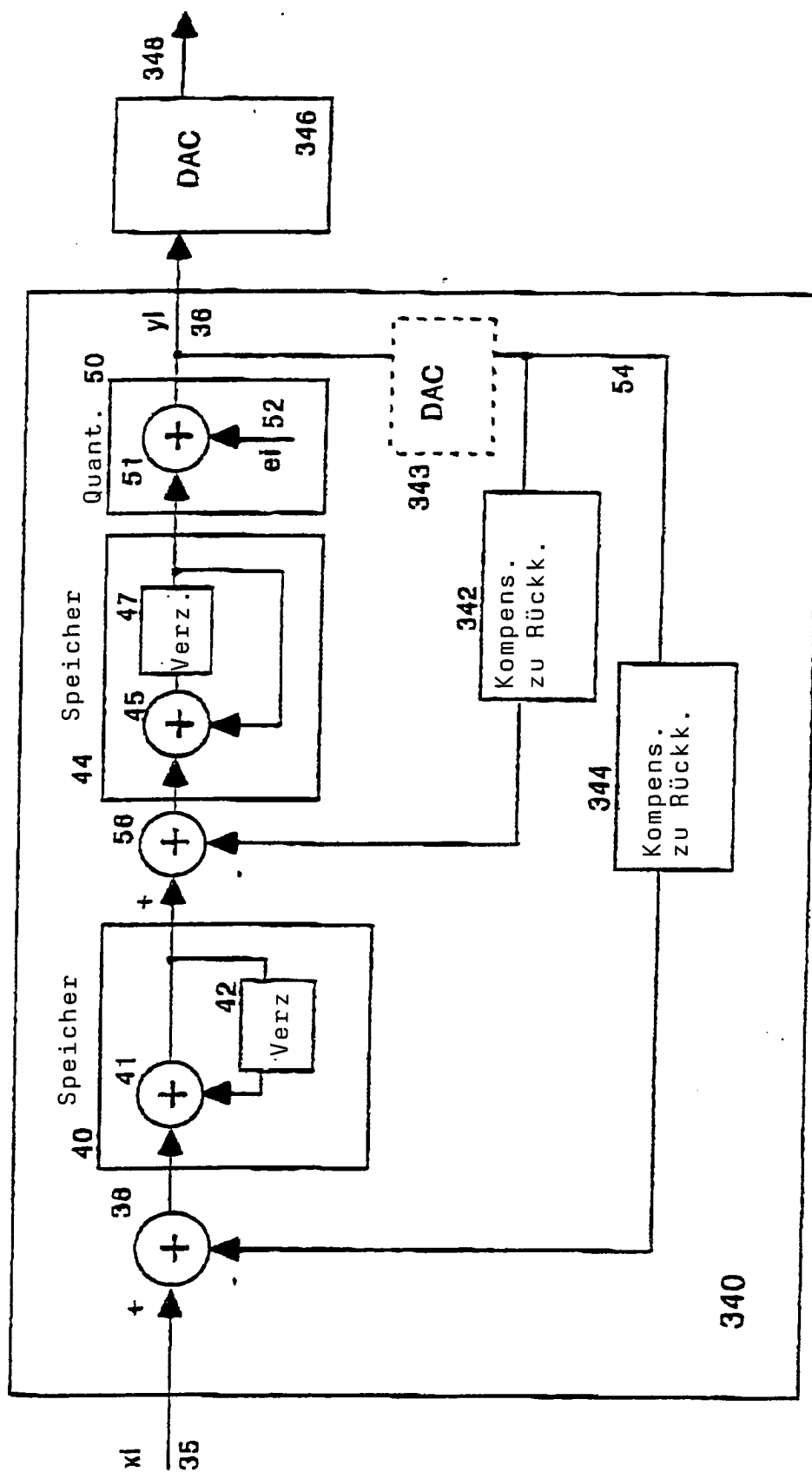
FIGURE 17E

[illegible]

FIGUR 17F

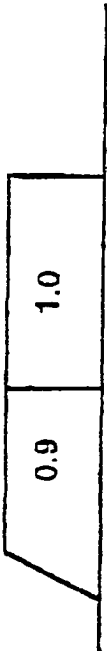


FIGUR 18



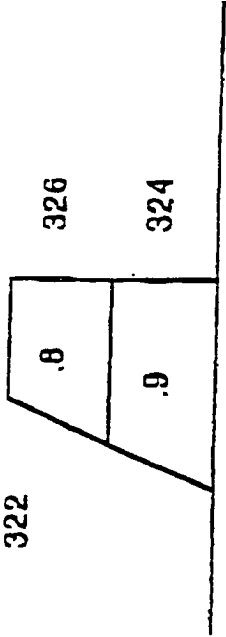
FIGUR 19

312

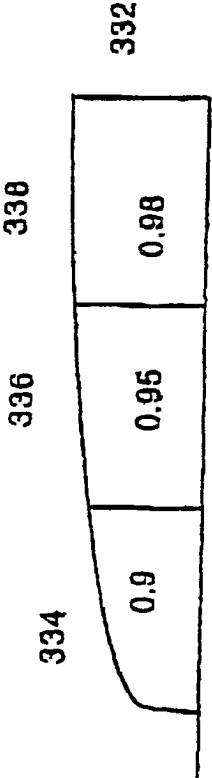


314 316

FIGUR 20



FIGUR 21



FIGUR 22

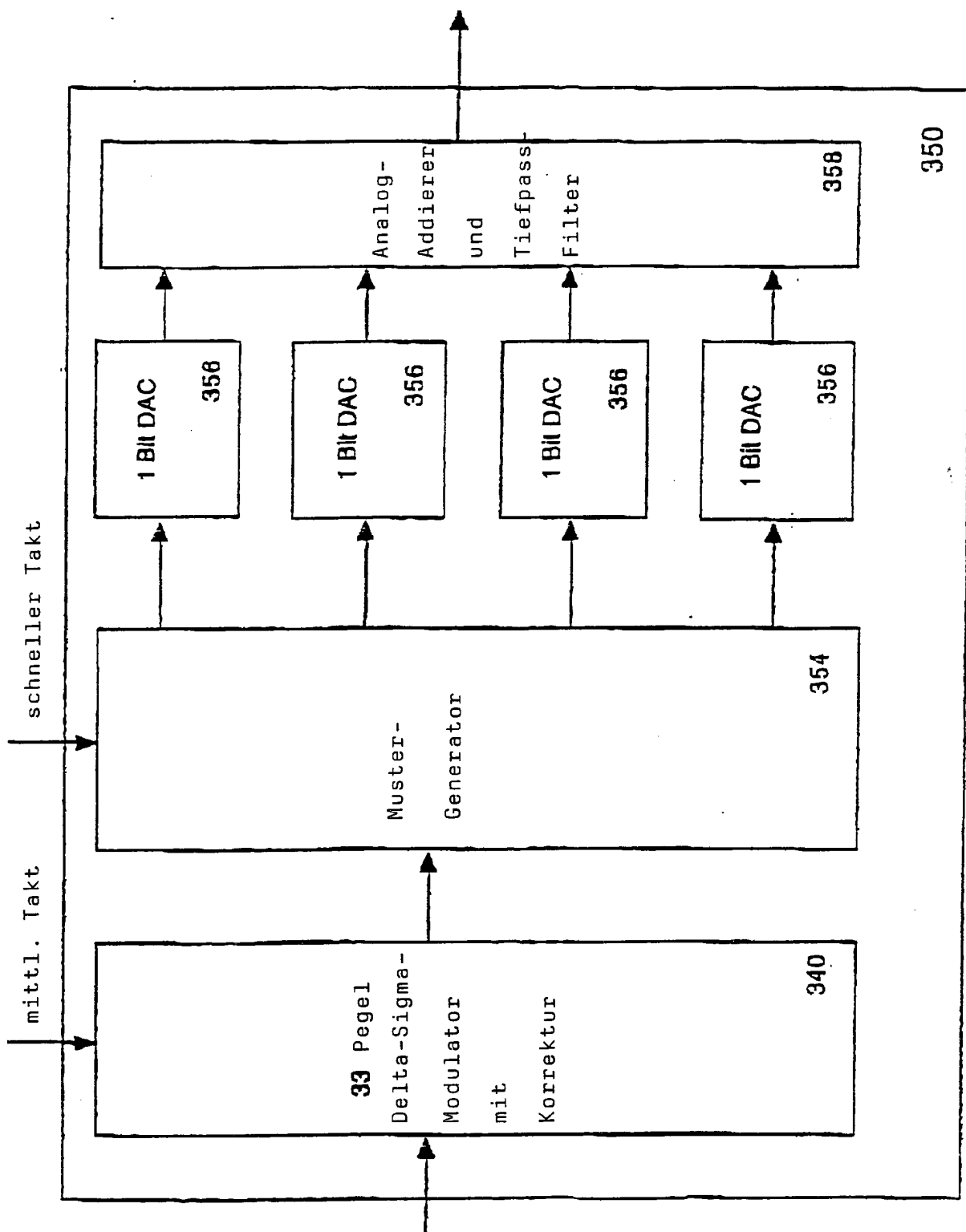
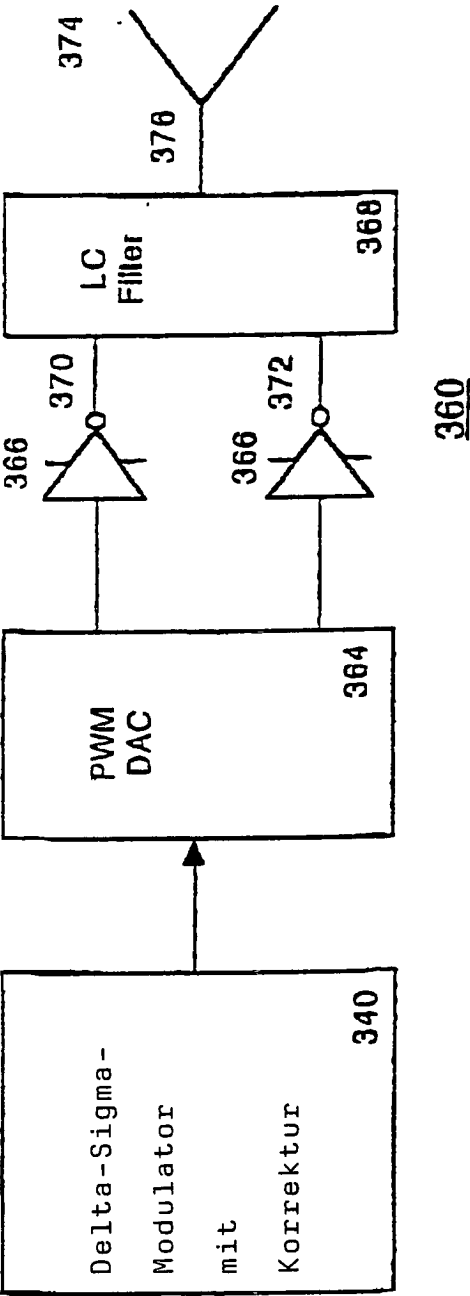
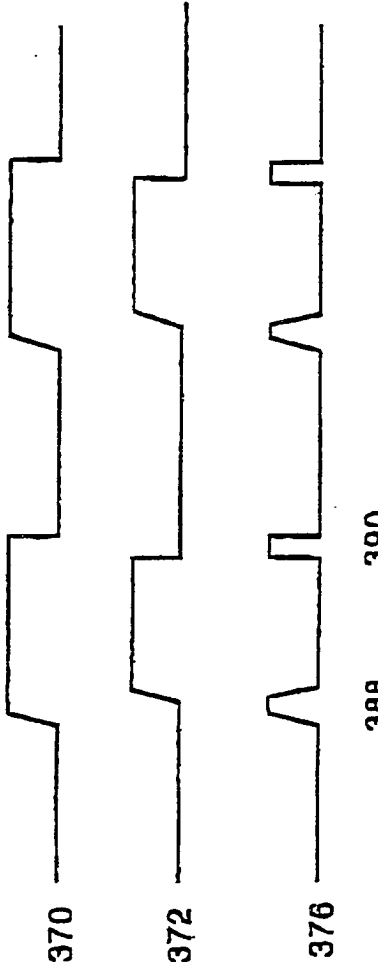


FIG. 23



FIGUR 24



FIGUR 25