



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) **DE 696 36 152 T2** 2007.05.03

(12) **Übersetzung der europäischen Patentschrift**

(97) **EP 0 724 363 B1**

(51) Int Cl.⁸: **H04N 7/24** (2006.01)

(21) Deutsches Aktenzeichen: **696 36 152.3**

(96) Europäisches Aktenzeichen: **96 300 495.7**

(96) Europäischer Anmeldetag: **24.01.1996**

(97) Erstveröffentlichung durch das EPA: **31.07.1996**

(97) Veröffentlichungstag

der Patenterteilung beim EPA: **24.05.2006**

(47) Veröffentlichungstag im Patentblatt: **03.05.2007**

(30) Unionspriorität:

3003295 25.01.1995 JP

(84) Benannte Vertragsstaaten:

DE, FR, GB

(73) Patentinhaber:

Sony Corp., Tokio/Tokyo, JP

(72) Erfinder:

Wada, c/o Intel. Prop. Div., Toru, Tokyo 141, JP

(74) Vertreter:

**Mitscherlich & Partner, Patent- und
Rechtsanwälte, 80331 München**

(54) Bezeichnung: **Vorrichtung zur Dekodierung eines digitalen Signals**

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

[0001] Die vorliegende Erfindung bezieht sich auf eine digitale Signaldecodiervorrichtung, insbesondere, jedoch nicht ausschließlich, auf eine digitale Signaldecodiervorrichtung, welche für eine Aufzeichnungs-/Wiedergabevorrichtung zum Aufzeichnen oder Reproduzieren eines Bewegtbildsignals auf oder von einem Aufzeichnungsträger verwendet werden soll, beispielsweise einer magneto-optischen Platte oder einem Magnetband, und für einen Empfänger eines Fernsehkonferenzsystems, um ein Bewegtbildsignal über einen Übertragungsweg zu übertragen oder zu empfangen.

[0002] Ein vorgeschlagenes System zum Übertragen eines Bewegtbildsignals an einen fernen Ort, beispielsweise bei einem Fernsehkonferenzsystem, einem visuellen Telefonsystem oder einem Rundfunksystem wurde ein Verfahren zum Komprimieren und Codieren eines Videosignals verwendet, wobei eine Leitungskorrelation oder eine Interrahmenkorrelation zwischen Videosignalen verwendet wird, um eine Übertragungsleitung effektiv zu nutzen. Unter Verwendung der Leitungskorrelation ist es beispielsweise möglich, ein Videosignal durch eine orthogonale Transformations-Codierverarbeitung (beispielsweise DCT (diskrete Kosinus-Transformation)) zu komprimieren. Außerdem ist es unter Verwendung von Interrahmenkorrelation möglich, das Videosignal weiter zu komprimieren.

[0003] Allgemein gibt es keine sehr große Änderung zwischen Bildern von zeitlich benachbarten Rahmen. Das heißt, wenn der Unterschied zwischen diesen berechnet wird, ist das Differenzsignal klein. Daher wird das Differenzsignal codiert und der codierte Wert wird komprimiert. Durch das Übertragen lediglich des Differenzsignals kann jedoch das Ursprungsbild nicht wiederhergestellt werden. Daher wird ein Verfahren verwendet, welches ein Videosignal durch Umsetzen des Bilds jedes Rahmens in eines von drei Rahmenformaten komprimiert und codiert, d.h., das I-Bild, das P-Bild und das B-Bild.

[0004] [Fig. 1A](#) und [Fig. 1B](#) zeigen das obige Codierverfahren. Im Fall dieses Kompressions- und Codierverfahrens wird eine Reihe von Rahmengruppen für jeweils 17 Rahmen (Rahmen F1 bis F17) verarbeitet. Diese Verarbeitungseinheit wird als eine Bildgruppe bezeichnet. Die Bildgruppe wird in ein I-Bild, B-Bild und P-Bild in der Reihenfolge mit Beginn mit dem ersten Rahmen F1 codiert, und der vierte und die nachfolgenden Rahmen F4 bis F17 werden abwechselnd in das B-Bild oder P-Bild codiert.

[0005] In diesem Fall ist das I-Bild das Bild, welches durch unmittelbares Codieren der Videosignale für einen Rahmen erlangt wird. Wie außerdem in [Fig. 1A](#) gezeigt ist, ist das P-Bild grundsätzlich das Bild, wel-

ches durch Codieren der Differenz zwischen den Videosignalen zum I-Bild, welches zeitlich dem P-Bild vorhergeht, oder der Differenz zwischen den Videosignalen zum zeitlich vorhergehenden P-Bild erlangt wird. Außerdem ist, wie in [Fig. 1B](#) gezeigt ist, das B-Bild grundsätzlich das Bild, welches durch Codieren der Differenz zwischen den Videosignalen zum Durchschnittswert zwischen dem zeitlich vorhergehenden Rahmen und einem zeitlich nachfolgenden Rahmen erlangt wird. Dieses Codierverfahren wird als bidirektionales Vorhersagecodieren bezeichnet.

[0006] In diesem Zusammenhang verwendet das B-Bild die folgenden drei Codierverfahrensarten zusätzlich zur bidirektionalen Vorhersagecodierung. Beim ersten Verarbeitungsverfahren werden unmittelbar die Daten für den ursprünglichen Rahmen F2 als Übertragungsdaten übertragen. Dies wird als Intra-Codierung bezeichnet, was die gleiche Verarbeitung wie im Fall des I-Bilds ist. Das zweite Verarbeitungsverfahren berechnet und überträgt eine Differenz von dem zeitlich folgenden Rahmen F3. Dies wird als Rückwärts-Vorhersagecodierung bezeichnet. Das dritte Verarbeitungsverfahren überträgt eine Differenz von dem zeitlich-vorhergehenden Rahmen F1. Dies wird als Vorwärts-Vorhersagecodierung bezeichnet.

[0007] Im Zeitpunkt der Codierung werden die Daten, welche durch das Verfahren zum Minimieren der Übertragungsrate unter diesen vier Codierverfahren codiert werden, als B-Bild verwendet.

[0008] Eine aktuelle Codiervorrichtung setzt außerdem die Videosignale des Rahmenformats (I-Bild, P-Bild oder B-Bild) in Blockformatsignale um und überträgt diese in Form von Bitströmen.

[0009] [Fig. 2A](#) bis [Fig. 2C](#) zeigen das Blockformat. Wie in [Fig. 2A](#) gezeigt ist, sind V Zeilen der Zeile, welche aus H Punkten für eine Zeile zusammengesetzt ist, zu einem Rahmenformat-Videosignal zusammengesetzt.

[0010] Das Videosignal eines Rahmens ist in N Slices unterteilt, von denen jedes 16 Zeilen aufweist. Jedes Slice besteht aus M Makroblöcken. Jeder Makroblock besteht aus einem Luminanzsignal, welches 16×16 Bildelementen (Punkten (dots)) entspricht, und das Luminanzsignal ist in Blöcke Y[1] bis Y[4] unterteilt, von denen jeder 8×8 Punkte aufweist. Das Luminanzsignal von 16×16 Punkten entspricht Farbsignalen Cb und Cr von 8×8 Punkten.

[0011] Die Decodiervorrichtung empfängt und decodiert somit das Videosignal, welches in einen Bitstrom umgesetzt wurde, über einen Aufzeichnungsträger oder eine Übertragungsleitung. Wenn ein Fehler im Bitstrom enthalten ist, der von einem Aufzeichnungsträger oder dgl. gelesen wird, kann dieses Feh-

lerteil auf ein Wiedergabebild oder dgl. reflektiert werden. Daher sind, wie in [Fig. 3](#) gezeigt ist, eine Fehlerermittlungs- und Korrekturcode-Additionsschaltung **1A** und eine Fehlerermittlungs- und Korrekturschaltung **2A** üblicherweise für eine Codiervorrichtung **1** bzw. eine Decodiervorrichtung **2** vorgesehen, so dass ein kleiner Fehler korrigiert werden kann.

[0012] Umso mehr die Länge des Fehlerkorrekturcodes, der hinzugefügt wird, vergrößert wird, desto mehr kann die Fehlerkorrekturkapazität um einen Wert äquivalent zur vergrößerten Länge vergrößert werden. Da jedoch die Menge von Übertragungsdaten ansteigt, wenn die Codelänge ansteigt, wird normalerweise ein Fehlerkorrekturcode mit einer passenden Länge verwendet. Wenn daher ein großer Datenfehler in einem Bitstrom S2 auftritt, kann der Fehler durch die Fehlerermittlungs- und Korrekturschaltung **2A** in einigen Fällen nicht korrigiert werden. In diesem Fall besteht ein Problem, dass nicht vermieden werden kann, dass ein Fehler, der im Bitstrom S2 enthalten ist, sich beim reproduzierten Bild widerspiegelt.

[0013] Obwohl dies die Probleme sind, die im Zeitpunkt normaler Reproduktion verursacht werden, gibt es außerdem die folgenden Probleme im Zeitpunkt einer Spezialreproduktion, beispielsweise einer schnellen Vorlaufreproduktion oder einer schnellen Rücklaufreproduktion eines Bilds zusätzlich zu den obigen Problemen. Das heißt, im Zeitpunkt einer Spezialreproduktion kann die Übertragungseinheit (Zugriffseinheit der Platte, usw.) in der Mitte eines Makroblocks enden. In diesem Fall wird das Verfahren verwendet, bei dem ein Fehlerstartcode D_{ES} einschließlich eines Synchroncodes in den Bitdatenstrom S2 eingefügt wird, so dass eine Decodierschaltung den Endpunkt im Makroblock identifizieren kann. Das folgende Problem kann jedoch im Makroblock einschließlich des Fehlerstartcodes D_{ES} auftreten.

[0014] Im Allgemeinen arbeiten ein inverser Quantisierer, der eine Decodierschaltung bildet, und eine IDCT-Schaltung (inverse diskrete Kosinus-Transformationsschaltung) in Blockeinheiten, und eine Bewegungskompensationsschaltung arbeitet in Makroblöcken. Außerdem wird die Beschreibung anschließend unter der Annahme ausgeführt, dass der Bitstrom S2 einschließlich eines nichtvollständigen Makroblocks einer variablen Längen-Decodierschaltung zugeführt wird, die in der Eingangsstufe der Decodierschaltung angeordnet ist.

[0015] Wie in [Fig. 4A](#) gezeigt ist, sind im Fall eines Makroblock "1" Makroblockdaten in Blöcken zwischen "1-1" und "1-6" vollständig. Im Fall eines Makroblocks "2" ist ein Block "2-2" unvollständig und durch den Fehlerstartcode D_{ES} unterbrochen.

[0016] In diesem Fall liefert für den Makroblock "1", bei dem ein Bitstrom bis zum Ende vorhanden ist, die variable Längendecodierschaltung ein Rahmen/Feld-DCT-Flag, ein Quantisierungsmaß, einen Vorhersagemodus, einen Bewegungsvektor und ein Rahmen/Feldvorhersageflag zu den Schaltungen in der hinteren Stufe nach Abschluss der Verarbeitung des Datenkopfs des Makroblocks "1", wie in [Fig. 4C](#) gezeigt ist. Danach liefert, wie in [Fig. 4D](#) gezeigt ist, die variable Längendecodierschaltung Bilddaten zur inversen Quantisierschaltung, wenn die Verarbeitung jedes Blocks endet.

[0017] Wenn dagegen der Bitstrom begrenzt ist, wie beim Makroblock "2" gezeigt ist, ermittelt die variable Längendecodierschaltung die Begrenzung des Bitstroms durch den Fehlerstartcode D_{ES} einschließlich eines Synchroncodes. Da jedoch der Makroblock "2" unvollständig ist, ist es unmöglich, die Verarbeitung des Blocks "2-2" zu beenden. Außerdem ist es unmöglich, die Verarbeitung des Makroblocks "2" zu beenden.

[0018] Da jedoch die inverse Quantisierschaltung und die IDCT-Schaltung blockweise arbeiten, werden deren Arbeitsweisen beeinträchtigt, wenn die Verarbeitung des Blocks "2-2" nicht beendet wird. Da außerdem die Bewegungskompensationsschaltung in jedem Makroblock arbeitet, kann die Arbeitsweise der Schaltung beeinträchtigt werden, wenn die Verarbeitung des Makroblocks "2" nicht abgeschlossen ist.

[0019] In diesem Fall wird das Verfahren in Betracht gezogen, bei dem eine Speicherschaltung zum Speichern der Verarbeitungsergebnisse der variablen Längendecodierschaltung für einen Makroblock vorgesehen ist und danach die Daten zu einer Schaltung in einer hinteren Stufe geliefert werden, wenn die Verarbeitung für einen Makroblock vollständig endet. Unter Verwendung dieses Verfahrens ist es möglich, zu verhindern, dass die hintere Stufe der Schaltung ausfällt, wenn keine Daten zur hinteren Stufe der Schaltung geliefert werden, sogar, wenn ein nichtvollständiger Makroblock vorhanden ist.

[0020] Da jedoch dieses Verfahren eine Speicherschaltung für einen Block nach der variablen Längendecodierschaltung erfordert, kann nicht vermieden werden, dass sich ein Schaltungsaufwand vergrößert oder dass die Anzahl von Verarbeitungsschaltungen ansteigt.

[0021] Eine digitale Signalaufzeichnungs- und Wiedergabevorrichtung ist in der EP-A 0 624 977 beschrieben, welche das Vorhandensein eines nicht korrigierbaren Fehlers in Videodaten identifiziert, welche von einem Aufzeichnungsträger reproduziert werden und welche die Fehlerdaten durch Daten an einer entsprechenden Position in der Vergangenheit ersetzt.

[0022] Entsprechende Merkmale und Gesichtspunkte der Erfindung sind in den Ansprüchen definiert.

[0023] Ein nichtkorrigierbares Fehlerteil oder Datenteil einschließlich des Fehlerteils im Digitalsignal kann durch den Spezialcode einschließlich eines Synchroncodes ersetzt werden und an die Decodiereinrichtung ausgegeben werden. Der Ersatzspezialcode kann einen Synchroncode aufweisen und immer durch die Decodiereinrichtung ermittelt werden. Daher ist es möglich, Bestätigung eines Vorhandenseins eines Fehlers und eine Fehlerkorrektur im Zeitpunkt der Decodierung zu realisieren.

[0024] Wenn außerdem eine Datengruppe mit vorher festgelegte Länge, welche das Digitalsignal bildet, wegen dieser Begrenzung durch den vorher festgelegten Code unvollständig ist, gibt die Decodiereinrichtung das vorhergesagte Digitalsignal anstelle des decodierten Signals der vollständigen Datengruppe aus. Dadurch ist es möglich, ein Ausgangssignal zu erlangen, welches nicht durch die Daten in der nichtvollständigen Datengruppe beeinflusst wird, ohne eine Spezialschaltung hinzuzufügen, um die nichtvollständige Datengruppe zu verarbeiten.

[0025] Die Ausführungsformen der vorliegenden Erfindung beschreiben anschließend:
eine digitale Signaldecodiereinrichtung, die in der Lage ist, Fehler zu vermeiden, welche in einem Bitstrom enthalten sind, damit diese nicht auf einen Decodierbildschirm reflektiert werden; und
eine Digitalsignal-Decodiereinrichtung, die in der Lage ist, Decodieren fortzusetzen, ohne irgendwelche Schaltungen sogar für einen nichtvollständigen Makroblock hinzuzufügen.

[0026] Die Erfindung wird weiter mittels eines beispielhaften und nichteinschränkenden Ausführungsbeispiels mit Hilfe der beiliegenden Zeichnungen beschrieben, wobei gleiche Teile mit gleichen Bezugszeichen bezeichnet sind, und in denen:

[0027] [Fig. 1A](#) und [Fig. 1B](#) schematische Diagramme sind, welche Bildarten erläutern, die verwendet werden, Bilddaten zu komprimieren;

[0028] [Fig. 2A](#) bis [Fig. 2C](#) schematische Diagramme sind, um Datenstrukturen von Bilddaten zu erläutern;

[0029] [Fig. 3](#) ein Blockdiagramm ist, welches ein Beispiel eines vorher vorgeschlagenen Videosignal-Codiergeräts und einer Decodiereinrichtung zeigt;

[0030] [Fig. 4A](#) bis [Fig. 4D](#) Zeitablaufdiagramme sind, welche Verarbeitungsarten durch eine herkömmliche variable Längendecodierschaltung zei-

gen;

[0031] [Fig. 5](#) ein Blockdiagramm ist, welches eine Ausführungsform der Digitalsignal-Decodiereinrichtung nach der vorliegenden Erfindung zeigt;

[0032] [Fig. 6](#) ein Blockdiagramm ist, welches eine weitere Ausführungsform der Digitalsignal-Decodiereinrichtung nach der vorliegenden Erfindung zeigt;

[0033] [Fig. 7](#) ein Flussdiagramm ist, welches die Verarbeitungsprozedur einer variablen Längendecodierschaltung erläutert;

[0034] [Fig. 8](#) ein Flussdiagramm ist, welches ein Beispiel der Verarbeitungsprozedur zeigt, welche für einen Normalbetrieb verwendet wird;

[0035] [Fig. 9](#) ein Flussdiagramm ist, welches ein Beispiel der Verarbeitungsprozedur zeigt, die für Spezialreproduktion oder Datenverarbeitung einschließlich eines nichtkorrigierbaren Fehlers verwendet wird;

[0036] [Fig. 10](#) ein Flussdiagramm ist, welches ein Beispiel der Verarbeitungsprozedur zeigt, welche für Spezialreproduktion oder Datenverarbeitung einschließlich eines nichtkorrigierbaren Fehlers verwendet wird; und

[0037] [Fig. 11A](#) bis [Fig. 11H](#) Zeitablaufdiagramme sind, welche Verarbeitungsarten durch eine variable Längendecodierschaltung zeigen.

[0038] Bevorzugte Ausführungsformen der vorliegenden Erfindung werden mit Hilfe der beiliegenden Zeichnungen beschrieben:

(1) Erste Ausführungsform

(1-1) Gesamtaufbau

[0039] [Fig. 5](#) zeigt den Schaltungsaufbau einer Decodiereinrichtung **12**. Die Decodiereinrichtung **12** besitzt einen Decodierschaltungsabschnitt **13**, um zu verhindern, dass ein Fehler auf ein Ausgangsbild reflektiert wird, wobei ein Fehlerteil ersetzt wird, welches nicht ausreichend durch eine Fehlerermittlungs-Korrekturschaltung mit dem Fehlerstartcode D_{ES} einschließlich eines Synchroncodes korrigiert werden kann.

[0040] Der Decodierschaltungsabschnitt **13** ist mit einer Fehlerermittlungs- und Korrekturschaltung **14**, einer Nichtkorrekturflag-Umsetzungsschaltung **15** und einer Decodierschaltung **16** ausgestattet. Jede dieser Schaltungen wird anschließend ausführlich beschrieben.

[0041] Die Fehlerermittlungs- und Korrekturschal-

tung **14** ermittelt das Vorhandensein oder Nichtvorhandensein eines Fehlers, der in einem Aufzeichnungsträger **3** oder einem Bitstrom S2 vorhanden ist, der über einen Übertragungsweg zugeführt wird, und korrigiert den Fehler unter Verwendung eines Fehlerkorrekturcodes, wenn ein Fehler ermittelt wird. Die Fehlerermittlungs- und Korrekturschaltung **14**, welche bei dieser Ausführungsform verwendet wird, gibt ein Nichtkorrekturflag D_{EF} zusammen mit einem Bitstrom D_{BS} aus und informiert eine Schaltung einer hinteren Stufe über ein nichtkorrigierbares Fehlerteil.

[0042] Die Nichtkorrekturflag-Umsetzungsschaltung **15** liefert den Bitstrom D_{BS} und das nichtkorrigierbare Flag D_{EF} und ersetzt ein Fehlerteil, welches nicht ausreichend durch die Fehlerermittlungs- und Korrekturschaltung **14** mit dem Fehlerstartcode D_{ES} korrigiert werden kann. Insbesondere empfängt eine Fehlerstartcode-Erzeugungsschaltung **15A** das nichtkorrigierbare Flag D_{EF} , um ein Fehlerteil zu ermitteln, welches nicht ausreichend durch die Ermittlungs- und Korrekturschaltung **14** korrigiert werden kann, erzeugt den Fehlerstartcode D_{ES} , wenn das Fehlerteil ermittelt wird, und gibt den Code D_{ES} an eine Schaltschaltung **15B** aus. Gemäß dem Fehlerstartcode D_{ES} ersetzt die Schaltschaltung **15B** ein Fehlerteil, welches im Bitstrom S2 enthalten ist, oder ein Datenteil, welches das Fehlerteil hat und die gleiche Datenlänge wie den Fehlerstartcode D_{ES} hat, durch den Fehlerstartcode D_{ES} .

[0043] Wenn übrigens das Fehlerteil, welches nicht ausreichend durch die Fehlerermittlungs- und Korrekturschaltung **14** korrigiert werden kann, bezüglich der Länge den Fehlerstartcode übersteigt, werden mehrere Fehlerstartcodes ($k \times D_{ES}$) unmittelbar auf das Fehlerteil geschrieben.

[0044] Bei dieser Ausführungsform besteht der Fehlerstartcode D_{ES} aus einem 32-Bit-Code, der dargestellt wird durch: "00000000 00000000 00000000 00000001 10110100". Der obige Code ist durch das Videosignal-Codiervorgehen, welches zur Zeit allgemein verwendet wird, definiert, bei dem die anfänglichen 24 Bits von 32 Bits als Synchroncode dienen. Außerdem dienen die abschließenden 8 Bits als Attributcode, die identifizieren, was der Synchroncode zeigt. In diesem Fall dienen die Bits „10110100“ als Attributcode, die einen Fehler zeigen.

[0045] In diesem Zusammenhang muss dieser Code nicht für irgendeine Kombination von Codes im variablen Längencodier-Bitstrom erscheinen und kann unter irgendeinem Decodierzustand in der Decodierschaltung **16** gefunden werden.

[0046] Damit ersetzt die Nichtkorrekturflag-Umsetzungsschaltung **15** das Fehlerteil im Bitstrom und das Teil einschließlich des Fehlerteils und entsprechend der Länge des Fehlerstartcodes D_{ES} durch den Fehlerstartcode D_{ES} .

Der Grund dafür liegt darin: wenn der Fehlerstartcode D_{ES} lediglich in das Fehlerteil eingefügt wird, vergrößert sich die Datenmenge des Bitstroms um die Einfügung des Fehlerstartcodes D_{ES} im Vergleich zu der des Bitstroms, der durch eine Codiervorrichtung erzeugt wird, um einen Wert, der einem Fehlerstartcode D_{ES} äquivalent ist. Da im Allgemeinen die Decodiervorrichtung ein System zum Steuern der Erzeugung von Bitströmen ist, während die Kapazität eines Empfangspuffers geprüft wird, kann die Möglichkeit, dass der Empfangspuffer ausfällt, nicht vermieden werden, wenn die Datenmenge ansteigt, wie oben beschrieben.

[0047] Wenn außerdem ein Flag dem Fehlerteil hinzugefügt wird, wird ein Puffer zum Speichern des Flags benötigt, um das Flag einem Bitstrom hinzuzufügen, um somit ausreichend das Fehlerteil zu überdecken. Daher besteht das Problem darin, dass die Kapazität des Puffers ansteigt.

[0048] Wenn jedoch der Bitstrom D_{BS} durch den Fehlerstartcode D_{ES} ersetzt wird, wie bei dieser Ausführungsform gezeigt ist, werden Bitströme gleich der Anzahl von Bitströmen, die durch die Codiervorrichtung erzeugt werden, im Pufferspeicher gespeichert, und es ist möglich, zu verhindern, dass der Puffer ausfällt.

[0049] Außerdem ersetzt bei dieser Ausführungsform das nichtkorrigierbare Flag D_{EF} durch den Fehlerstartcode D_{ES} , unmittelbar vor dem Empfangspuffer (**17** in [Fig. 6](#)), um zu verhindern, dass die Kapazität des Empfangspuffers ansteigt, indem der Bitstrom D_{BS} und außerdem das nichtkorrigierbare Flag gespeichert wird.

[0050] Die Decodierschaltung **16** wird anschließend beschrieben. Die Decodierschaltung **16** liefert den Bitstrom S3, der von der Nichtkorrekturflag-Umsetzungsschaltung **15** über einen Empfangspuffer ausgegeben wird, um verschiedene Decodierverarbeitungen auszuführen: variable Längendecodierverarbeitung; inverse Quantisierungsverarbeitung; inverse diskrete Kosinus-Transformationsverarbeitung; und Bewegungskompensationsverarbeitung, usw..

[0051] Wenn der Fehlerstartcode D_{ES} vom Bitstrom S3 unterschieden wird, arbeitet die Decodierschaltung **16** so, dass ein Bild einschließlich Fehlern nicht vom Ausgangsanschluss gemäß einer Reihe von Decodierverarbeitungen nicht ausgegeben wird.

(1-2) Kurzbeschreibung der Decodierverarbeitungen

[0052] Für den obigen Aufbau wird eine Reihe von Decodierverarbeitungen durch die Decodiervorrichtung **12** anschließend beschrieben. Zunächst korrigiert die Decodiervorrichtung **12** Codefehler, welche im Bitstrom S2 existieren, durch die Fehlerermitt-

lungs- und Korrekturschaltung **14**, welche in der ersten Stufe vorgesehen ist. In diesem Fall ersetzt, wenn es ein Fehlerteil gibt, welches nicht ausreichend durch den Fehlerkorrekturcode korrigiert werden kann, die Fehlerkorrekturschaltung **14** Daten vorher festgelegter Länge einschließlich des Fehlerteils durch den Fehlerstartcode D_{ES} , um den Code zur Decodierschaltung **16** zu liefern. In diesem Fall decodiert der Decodierschaltungsabschnitt **13** den Bitstrom $S3$ in der Reihenfolge gemäß einer vorher festgelegten Decodierprozedur. Wenn jedoch der Fehlerstartcode D_{ES} enthalten ist, bestätigt der Decodierschaltungsabschnitt **13** die Position des Codeteils und wählt unterschiedliche Verarbeitung aus, so dass sich kein Fehler auf ein decodiertes Bild widerspiegelt.

[0053] Gemäß dem obigen Aufbau wird das Datenteil einschließlich des nichtkorrigierbaren Datenfehlers durch den Fehlerstartcode D_{ES} ersetzt, so dass der Code durch die Decodierschaltung **13** unterschieden werden kann, so dass es möglich ist, die Verarbeitung auszuwählen, um zu verhindern, dass Datenfehler auftreten, welche nicht ausreichend durch die Fehlerermittlungs- und Korrekturschaltung **14** korrigiert werden können, dass diese auf das decodierte Bild reflektiert werden. Dadurch ist es möglich, die Bilddaten-Decodiervorrichtung zu realisieren, mit der ein Bild mit höherer Qualität als üblich erzielt werden kann.

(2) Ausführliche Beschreibung der Decodierschaltung

(2-1) Gesamtaufbau

[0054] Ein Schaltkreisbeispiel der Decodierschaltung **16**, welche im vorhergehenden Abschnitt beschrieben wurde, wird hier beschrieben. Die Decodierschaltung **16** ermöglicht es, das Decodieren fortzusetzen, ohne die Verarbeitung zu beeinträchtigen, wobei ein vorhergehendes Bild für den Teil eines nichtvollständigen Makroblocks ersetzt wird, sogar, wenn der Makroblock im Zeitpunkt von Spezialreproduktion auftritt (beispielsweise im Zeitpunkt einer schnellen Vorlaufreproduktion oder einer schnellen Umkehrreproduktion). In diesem Fall wird außerdem, da das vorhergehende Bild durch den nichtvollständigen Makroblock ersetzt ist, verhindert, dass der nichtvollständige Makroblock sich auf andere Bilder widerspiegelt.

[0055] [Fig. 6](#) zeigt ein Beispiel der Decodierschaltung **16**. Die Decodierschaltung **16** fängt den Bitstrom $S3$ von Bilddaten, welche über die Fehlerermittlungs- und Korrekturschaltung **14** und die Nichtkorrekturflag-Umsetzungsschaltung **15** zugeführt werden, im Empfangspuffer **17** ein, um vorübergehend den Bitstrom $S3$ zu speichern.

[0056] Eine variable Längendecodierschaltung **18** ist eine Hauptschaltung dieser Ausführungsform, welche den Quantisierungsschritt $S4$, den Bewegungsvektor $S5$, den Vorhersagemodus $S6$ (Information in Bezug auf die Vorhersage im Makroblock), das Rahmen-/Feld-Vorhersageflag $S7$ (anschließend als Vorhersageflag $S7$ bezeichnet) und das Rahmen-/Feld-DCT-Flag $S8$ (anschließend als DCT-Flag $S8$ bezeichnet) durch variables Längendecodieren der Datengruppe, welche vom Empfangspuffer **17** gelesen wird, verschafft.

[0057] Die variable Längendecodierschaltung **18** steuert eine Verarbeitungsschaltung einer hinteren Stufe, indem der Quantisierungsschritt $S4$ unter diesen unterschiedlichen Arten an Decodierinformation der inversen Quantisierschaltung **19** bereitgestellt wird, und wobei der Bewegungsvektor $S5$, der Vorhersagemodus $S6$, das Vorhersageflag $S7$ und das DCT-Flag **8** dem Bewegungskompensations-Schaltungsabschnitt **20** zugeführt werden. Die variable Längendecodierschaltung **18** liefert außerdem ein DCT-Ausgangsdaten-Maskierungsflag $S9$ (anschließend als Master-Flag $S0$ bezeichnet), um zu verhindern, dass die entsprechenden Bilddaten im Ermittlungszeitpunkt des Fehlerstartcodes D_{ES} auf ein nichtkorrigierbares Fehlerteil geschrieben werden, um nicht von der Ausgangsendstufe an den Bewegungskompensations-Schaltungsabschnitt **20** ausgegeben zu werden.

[0058] Die inverse Quantisierschaltung **19** quantisiert die Bilddaten, welche durch die variable Längendecodierschaltung **18** decodiert werden, invers gemäß dem Quantisierungsschritt $S4$, der von der variablen Längendecodierschaltung **18** geliefert wird, und gibt die invers-quantisierten Daten an eine IDCT-Schaltung **21** aus. Die IDCT-Schaltung **21** führt die inversen DCT-Prozesse in Bezug auf die Daten (DCT-Koeffizienten) durch, welche von der inversen Quantisierschaltung **19** geliefert werden, und liefert das Verarbeitungsergebnis zu einer Rahmen-/Feld-DCT-Blockumordnungsschaltung **22**, welche die Anfangsstufe des Bewegungskompensations-Schaltungsabschnitts **20** bildet. Die Rahmen-/Feld-DCT-Blockumordnungsschaltung **22** ordnet die Daten entsprechend einem Bildformat um.

[0059] Eine Berechnungseinheit **23** führt die Maskierungsoperation von Bilddaten $S10$ aus, welche über die inverse Quantisierschaltung **19**, die IDCT-Schaltung **21** und die DCT-Blockumordnungsschaltung **22** zu einem Makroblock geführt werden, für den die Maske gemäß der Instruktion des Maskierungsflags $S9$ bestimmt ist. In diesem Fall wird die Maskierungsoperation als eine Verarbeitung definiert, um die Bilddaten $S10$ einem Vorhersagebild $S11$ nicht hinzuzufügen, welches von einer Bewegungskompensationsschaltung **24** geliefert wird, oder um die Bilddaten $S10$ als Null zum Bild $S11$ hin-

zuzufügen. Gemäß der obigen Maskierungsoperation wird lediglich ein Vorhersagebild von der Berechnungseinheit **23** als ein ausgegebenes Wiedergabebild S12 ausgegeben, sogar, wenn irgendeine Art von Bilddaten von der Rahmen-/Feld-DCT-Blockumordnungsschaltung **22** ausgegeben wird.

[0060] Die Bewegungskompensationsschaltung **24** erzeugt das Vorhersagebild S11 entsprechend dem Rahmenformat der Bilddaten S10, welche über die DCT-Blockumordnungsschaltung **22** geliefert werden, gemäß einem Bild, welches aus einem Rahmenspeicher **25** gelesen wird, und liefert das Bild S11 zur Berechnungseinheit **23**.

[0061] Wenn beispielsweise das I-Bild vorhanden ist, werden die Bilddaten S10 von der Berechnungseinheit **23** als Ausgabe-Wiedergabebild S12 ausgegeben und im Vorwärtsvorhersage-Bildabschnitt **25B** des Rahmenspeichers **25** gespeichert, um die Vorhersagebilddaten der Bilddaten zu erzeugen, welche danach zuzuführen sind (die Daten für das P- oder B-Bild).

[0062] Wenn die Bilddaten S10, die zugeführt werden, die Daten für das P-Bild sind, wobei Bilddaten eines Rahmens vor den Daten S10 als Vorhersagebilddaten verwendet werden und diese die Daten außerdem für einen Vorhersage einen Vorwärtsvorhersagemodus sind, liest die Bewegungskompensationsschaltung **24** die Bilddaten (die Daten für das I-Bild) einen Rahmen vorher vom Vorwärtsvorhersage-Bildabschnitt **25B** des Rahmenspeichers **25** aus.

[0063] Die Bewegungskompensationsschaltung **24** bewegungs-kompensiert die Bilddaten, welche vom Rahmenspeicher **25** ausgelesen werden, gemäß dem Bewegungsvektor S5, der von der variablen Längendecodierschaltung **18** geliefert wird, und gibt diese als Vorhersagebilddaten S11 aus.

[0064] Die Berechnungseinheit **23** addiert die Vorhersagebilddaten S11 und die Bilddaten (Differenzdaten) S10, die von der IDCT-Schaltung **21** geliefert werden, und gibt die Additionsausgangsdaten als Ausgangswiedergabebild S12 aus.

[0065] Die Additionsausgangsdaten, d.h., die Bilddaten des decodierten P-Bilds, werden in einem Rückwärtsvorhersage-Bildabschnitt **25A** des Rahmenspeichers **25** gespeichert, um Vorhersagebilddaten der Bilddaten (Daten für das B-Bild oder das P-Bild) zu erzeugen, die anschließend zugeführt werden. In diesem Zusammenhang gibt, wenn die Bilddaten für das P-Bild im Inter-Bild-Vorhersagemodus codiert sind, die Berechnungseinheit **23** unmittelbar die Bilddaten ähnlich den Bilddaten des I-Bilds aus. Außerdem werden die Bilddaten im Rückwärtsvorhersage-Bildabschnitt **25B** gespeichert.

[0066] In diesem Fall wird das P-Bild nicht an eine Formatumsetzungsschaltung (nicht gezeigt) in der hinteren Stufe ausgegeben, da dies ein Bild ist, welches im Anschluss an das B-Bild angezeigt wird.

[0067] Wenn das B-Bild, welches anschließend zugeführt wird, im Vorwärtsvorhersagemodus codiert ist, liest die Bewegungskompensationsschaltung **24** die Bilddaten für das I-Bild vom Vorwärtsvorhersage-Bildabschnitt **25B** entsprechend dem Vorhersagemodus S6 aus und bewegungs-kompensiert die Bilddaten durch den Bewegungsvektor S5, um somit das Vorhersagebild S11 zu erzeugen.

[0068] Wenn dagegen das B-Bild im Rückwärtsvorhersagemodus codiert ist, liest die Bewegungskompensationsschaltung **24** die Bilddaten für das P-Bild vom Rückwärtsvorhersage-Bildabschnitt **25A** entsprechend dem Vorhersagemodus S6 aus und bewegungs-kompensiert die Bilddaten durch den Bewegungsvektor S5, um somit ein Vorhersagebild SV zu erzeugen.

[0069] Wenn außerdem das B-Bild im bidirektionalen Vorhersagemodus codiert ist, liest die Bewegungskompensationsschaltung **24** die Bilddaten für das I-Bild und das P-Bild vom Vorwärtsvorhersage-Bildabschnitt **25B** und dem Rückwärtsvorhersage-Bildabschnitt **25A** entsprechend dem Vorhersagemodus S6 aus und bewegungs-kompensiert die Bilddaten durch den Bewegungsvektor S5, um das Vorhersagebild S11 zu erzeugen.

[0070] Damit werden bewegungs-kompensierte Bilddaten von der Bewegungskompensationsschaltung **24** an die Berechnungseinheit **23** als Vorhersagebild S11 ausgegeben und zu den Ausgangsdaten der IDCT-Schaltung **21** hinzugefügt.

[0071] In diesem Fall sind jedoch die Additionsausgangsdaten, welche von der Berechnungseinheit **23** ausgegeben werden, die Bilddaten für das B-Bild, die aktuell nicht verwendet werden, um Vorhersagebilder anderer Bilder zu erzeugen. Daher werden die Additionsausgangsdaten nicht im Rahmenspeicher **25** gespeichert.

[0072] Nachdem diese Bilder für das B-Bild ausgegeben sind, liest die Bewegungskompensationsschaltung **24** die Bilddaten für das P-Bild vom Rückwärtsvorhersage-Bildabschnitt **25A** und liefert diese zur Berechnungseinheit **23**. Das P-Bild ist jedoch nicht bewegungskompensiert.

[0073] In diesem Zusammenhang sind Schaltungen, welche der Vorhersagemodus-Schalterschaltung auf Seiten der Codiervorrichtung und der DCT-Modus-Schalterschaltung entsprechen, nicht in der Decodierschaltung **16** gezeigt. Die Bewegungskompensationsschaltung **24** führt jedoch die Verarbeitung ent-

sprechend den obigen Schaltungen aus, d.h., die Verarbeitung zum Rückstellen des Aufbaus, bei dem Zeilensignale gemäß der Notwendigkeit in ungeradzahligem oder geradzahligem Feldern voneinander getrennt sind, in den Ursprungszustand, in welchem die beiden Zeilensignale gemischt sind.

[0074] Obwohl oben die Verarbeitung eines Luminanzsignals beschrieben wurde, gilt diese ebenfalls für die Verarbeitung eines Farbdifferenzsignals. In diesem Fall wird jedoch ein Bewegungsvektor verwendet, der durch Halbieren eines Bewegungsvektors für ein Luminanzsignal vertikal und horizontal erlangt wird.

(2-2) Verarbeitung für jeden Makroblock durch die variable Längendecodierschaltung

(2-2-1) Überblick über die Verarbeitung

[0075] Die Verarbeitung durch die variable Längendecodierschaltung **18** wird anschließend mit Hilfe von [Fig. 7](#) bis [Fig. 10](#) beschrieben. Die variable Längendecodierschaltung **18** beginnt das Decodieren vom Schritt SP1 an. Im Schritt SP2 decodiert die variable Längendecodierschaltung **18** Makroblock-Datenköpfe von Bitströmen, die nacheinander zugeführt werden.

[0076] Danach läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP3, um zu bestimmen, ob der Fehlerstartcode D_{ES} im Datenkopfteil vorhanden ist oder nicht.

[0077] Wenn im Schritt SP3 ein bejahendes Ergebnis erlangt wird (d.h., wenn der Fehlerstartcode vorhanden ist), läuft die variable Längendecodierschaltung **18** unmittelbar weiter zum Schritt SP4, um die Verarbeitung des Makroblocks zu beenden.

[0078] Wenn jedoch ein negatives Ergebnis erlangt wird, (d.h., wenn der Fehlerstartcode nicht gefunden wird), läuft die variable Längendecodierschaltung **18** weiter zum Schritt SA5, um zu bestimmen, ob die Verarbeitung des Makroblocksdatenkopfs beendet ist oder nicht. Wenn ein negatives Ergebnis erlangt wird, kehrt die variable Längendecodierschaltung **18** zurück zum Schritt SP2 und wiederholt eine Reihe von Verarbeitungen.

[0079] Wenn dagegen ein zustimmendes Ergebnis erlangt wird (d.h., wenn die Verarbeitung des Makroblock-Datenkopfs beendet ist), läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP6, um das Quantisierungsmaß S4 und das DCT-Flag S8 zu erlangen und schreibt diese in die inverse Quantisierungsschaltung **19** und die Rahmen-/Feld-DCT-Blockumordnungsschaltung **22**.

[0080] Danach verarbeitet, wie im Schritt SP7 ge-

zeigt ist, die variable Längendecodierschaltung **18** jeden Block, der auf die Makroblock-Datenköpfe folgt, um Bilddaten für jeden Block zu erzeugen.

[0081] Außerdem läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP8, um zu bestimmen, ob der Fehlerstartcode D_{ES} enthalten ist oder nicht.

[0082] Wenn ein negatives Ergebnis im Schritt SP8 erlangt wird (d.h., wenn der Fehlerstartcode nicht gefunden ist), läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP9, um zu bestimmen, ob die Verarbeitung aller Blöcke beendet ist oder nicht und wiederholt eine Reihe von Verarbeitungen im Schritt SP7, bis die Verarbeitung aller Blöcke beendet ist (d.h., bis ein positives Ergebnis erlangt wird).

[0083] Wenn ein zustimmendes Ergebnis im Schritt SP9 erlangt wird, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP10, um die normale Hilfsroutinenverarbeitung auszuführen. Die Verarbeitung im Schritt SP10 wird ausführlich später beschrieben. Wenn die Verarbeitung endet, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP4, um die Verarbeitung des Makroblocks zu beenden.

[0084] Wenn dagegen ein zustimmendes Ergebnis im Schritt SP8 erlangt wird und bestimmt wird, dass der Fehlerstartcode im Makroblock enthalten ist, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP11, um Pseudodaten zu erzeugen, und steuert so, dass die Verarbeitung des Makroblocks durch Ausgeben der Pseudodaten als Bilddaten beendet wird. Jedoch, wenn die Pseudodaten im Schritt SP11 nicht erzeugt werden, braucht die Decodierschaltung **16** eine Menge an Zeit, um den Betriebszeitablauf einer Verarbeitungsschaltung der hinteren Stufe zu steuern, welche blockweise oder in Makroblöcken wie die inverse Quantisierungsschaltung **19** arbeitet, und es kann somit nicht vermieden werden, dass die Schaltung kompliziert wird.

[0085] Danach läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP12, um zu bestimmen, ob Pseudodaten entsprechend allen Blöcken erzeugt wurden oder nicht, und kehrt zum Schritt SP11 zurück, um die Erzeugung der Pseudodaten fortzusetzen, bis ein bejahendes Ergebnis erlangt wird.

[0086] Wenn dann ein bejahendes Ergebnis im Schritt SP12 erlangt wird, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP13, um die Hilfsroutinenverarbeitung eines Makroblocks auszuführen, der den Fehlerstartcode aufweist. Die Details der Hilfsroutinenverarbeitung werden ebenfalls später beschrieben. Wenn diese Verarbeitungen beendet sind, läuft die variable Längendecodierschal-

tung **18** weiter zum Schritt SP4, um die Verarbeitung des Makroblocks zu beenden.

(2-2-2) Hilfsroutinenverarbeitung, welche durch einen Makroblock ausgeführt wird, der keinen Fehlerstartcode aufweist

[0087] Anschließend wird die Verarbeitungsprozedur im Schritt SP10, welche für einen Normalbetrieb verwendet wird, mit Hilfe von [Fig. 8](#) ausführlich beschrieben.

[0088] Zunächst beginnt die variable Längendecodierschaltung **18** die Verarbeitung vom Schritt SP21 an. Danach bestimmt im Schritt SP22 die variable Längendecodierschaltung **18**, ob ein Vorhersagebild zu erlangen ist oder nicht, wobei der Vorhersagemodus S6, der Bewegungsvektor S5 und das Vorhersageflag S7 eines Makroblocks verwendet wird, der unmittelbar vor dem Makroblock angeordnet ist, der verarbeitet wird und der aktuell vorhergesagt wird.

[0089] Wenn hier ein negatives Ergebnis erlangt wird, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP23, um das DCT-Ausgangsdaten-Maskierungsflag auf "0" zu setzen, um die Ausgangsdaten von Bilddaten, welche von einem Bitstrom erlangt werden, zu bestimmen, und schreibt die Daten des Bitstroms in eine Verarbeitungsschaltung einer hinteren Stufe als den Vorhersagemodus S6, den Bewegungsvektor S5 und das Vorhersageflag S7. Danach beendet die variable Längendecodierschaltung **18** eine Reihe von Arbeiten im Schritt SP24.

[0090] Wenn dagegen ein bejahendes Ergebnis im Schritt SP22 erhalten wird, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP25, um die Verarbeitung, welche die gleiche ist wie im Schritt SP23, auszuführen.

[0091] Nachdem die Verarbeitung im Schritt SP25 endet, bestimmt die variable Längendecodierschaltung **18**, ob der Vorhersagemodus, der als Verarbeitung im Schritt SP26 bestimmt wurde, die Bilddaten beim vorhandenen Bild verwendet oder nicht. Wenn ein bejahendes Ergebnis erhalten wird, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP24, um die Verarbeitung zu beenden. Wenn dagegen ein negatives Ergebnis im Schritt SP26 erhalten wird, schreibt die variable Längendecodierschaltung **18** den Vorhersagemodus S6, den Bewegungsvektor S5 und das Vorhersageflag S7 in einen Speicher.

(2-2-3) Hilfsroutinen-Verarbeitung, welche durch einen Makroblock ausgeführt wird, der einen Fehlerstartcode aufweist

[0092] Anschließend wird die Verarbeitung in dem Zeitpunkt, wenn ein Makroblock durch den Fehler-

startcode D_{ES} ausgefallen ist, mit Hilfe von [Fig. 9](#) und [Fig. 10](#) beschrieben. Diese Verarbeitung ist die Verarbeitung, ein Vorhersagebild zu erzeugen, welches anstelle eines Bitstrombilds ausgegeben wird, und die folgenden vier Verfahren können als Erzeugungseinrichtung angesehen werden.

[0093] Das erste Verfahren ist in den Stufen SP32 und SP33 gezeigt, welches ein Vorhersagebild an der räumlichen gleichen Position wie das Vorwärtsbild verwendet.

[0094] Das zweite Verfahren ist in den Stufen SP35 bis SP37 gezeigt ist, welches ein Vorhersagebild an der räumlich gleichen Position wie ein Bild einen Rahmen vorher in der Anzeigzeit verwendet.

[0095] Das dritte Verfahren verwendet ein Vorhersagebild, welches unter Verwendung des Vorhersagemodus, des Bewegungsvektors und eines Vorhersageflags eines Makroblocks erlangt wird, bei dem die vorhergehende Vorhersage durchgeführt wird.

[0096] Das vierte Verfahren ist in den Schritten SP40 bis SP42 gezeigt, bei dem ein Vorhersagebild verwendet wird, welches unter Verwendung des Vorhersagemodus, des Bewegungsvektors und des Vorhersageflags eines verarbeiteten Makroblocks erlangt wird.

[0097] Die variable Längendecodierschaltung **18** beginnt die Verarbeitung vom Schritt SP31, um ein Vorhersagebild durch eines dieser vier Verfahren zu erzeugen. Im Schritt SP32 bestimmt die variable Längendecodierschaltung **18**, ob ein Vorhersagebild an der räumlich gleichen Position wie ein vorderes Bild zu verwenden ist oder nicht.

[0098] Wenn hier ein bejahendes Ergebnis erlangt wird, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP33, um das DCT-Flag S9 auf "1" zu setzen und unterbindet, dass ein Bild von einem Bitstrom ausgegeben wird, der aus Pseudodaten und dgl. zusammengesetzt ist. Außerdem setzt die variable Längendecodierschaltung **18** den Vorhersagemodus S6 auf den Vorwärtsvorhersagemodus und einen Bewegungsvektor auf 0. In diesem Zusammenhang wird das Vorhersageflag S7 auf ein Feld fixiert oder auf das gleiche gesetzt, wie eine Bildstruktur ist.

[0099] Wenn diese Verarbeitungen enden, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP34, um die Verarbeitung zu beenden.

[0100] Wenn dagegen ein negatives Ergebnis im Schritt SP32 erlangt wird, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP35, um zu bestimmen, ob ein Vorhersagebild durch das zweite Verfahren zu erzeugen ist oder nicht. Das heißt, die variable Längendecodierschaltung **18** bestimmt, ob

ein Vorhersagebild an der räumlich gleichen Position wie ein Bild eines Rahmens vorher in der Anzeigezeit zu verwenden ist oder nicht.

[0101] Wenn hier ein bejahendes Ergebnis erlangt wird, läuft die Verarbeitung der variablen Längendecodierschaltung **18** weiter zum Schritt SP36, um zu bestimmen, ob die Bildart des Bilds, zu dem der Makroblock gehört, das I-Bild, das P-Bild oder das B-Bild ist, deren vorhergehendes Bild nicht das B-Bild ist. Wenn in diesem Fall ein negatives Ergebnis erlangt wird (d.h., das Bild, welches dem Bild vorhergeht, welches zu decodieren ist, das B-Bild ist) kehrt die variable Längendecodierschaltung **18** zurück zum Schritt SP32, um die Bestimmungsverarbeitung wiederum zu versuchen.

[0102] Wenn dagegen eine zustimmendes Ergebnis erlangt wird, läuft die Verarbeitung der variablen Längendecodierschaltung **18** weiter zum Schritt SP37, um zu vermeiden, dass die Bilddaten, welche vom Bitstrom erlangt werden, ausgegeben werden, und setzt den Vorhersagemodus S6 auf die Seite der vorderen Vorhersage. Auch in diesem Fall wird der Bewegungsvektor S5 auf null gesetzt, da die Bilddaten in der räumlich gleichen Position verwendet werden. Außerdem wird das Vorhersageflag S7 auf ein Feld fixiert oder nutzt ein Flag, welches das gleiche wie eine Bildstruktur ist. Nachdem diese Verarbeitungen enden, wird die Verarbeitung im Schritt SP34 beendet.

[0103] Wenn jedoch ein negatives Ergebnis im Schritt SP35 erlangt wird, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP38, um zu bestimmen, ob ein Vorhersagebild durch das dritte Verfahren erzeugt werden soll oder nicht. Das heißt, dass die variable Längendecodierschaltung **18** bestimmt, ob ein Vorhersagebild, welches unter Verwendung des Vorhersagemodus, des Bewegungsvektors und des Vorhersageflags eines Makroblocks, in welchem die vorhergehende Vorhersage durchgeführt wird, zu verwenden ist oder nicht.

[0104] Wenn hier ein bejahendes Ergebnis erlangt wird, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP39, um den Vorhersagemodus S6, den Bewegungsvektor S5 bzw. das Vorhersageflag S7 auf den Vorhersagemodus, den Bewegungsvektor bzw. das Vorhersageflag, welches in einem Speicher gespeichert ist, zu setzen.

[0105] Wenn dagegen ein negatives Ergebnis im Schritt SP38 erlangt wird, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP40, um zu bestimmen, ob ein Vorhersagebild durch das vierte Verfahren zu erzeugen ist oder nicht; d.h., die variable Längendecodierschaltung **18** bestimmt, ob ein Vorhersagebild zu verwenden ist oder nicht, welches unter Verwendung des Vorhersagemodus des Bewe-

gungsvektors und des Vorhersageflags eines verarbeitenden Makroblocks erlangt wird.

[0106] Wenn ein bejahendes Ergebnis im Schritt SP40 erlangt wird, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP41, um zu bestimmen, ob der Vorhersagemodus in einem Bild ist oder nicht. Wenn im Schritt SP41 ein negatives Ergebnis erlangt wird, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP42, um den Vorhersagemodus S6, den Bewegungsvektor S5 und das Vorhersageflag S7, welches vom Datenkopf des verarbeitenden Makroblocks gelesen wird, zu setzen.

[0107] Wenn eine Reihe dieser Verarbeitungen endet, läuft die variable Längendecodierschaltung **18** weiter zum Schritt SP34, um die Verarbeitung des vorhandenen Makroblocks zu beenden.

(2-3) Decodierbeispiel

[0108] In diesem Abschnitt wird ein spezielles Beispiel einer Verarbeitung der Decodierschaltung **16** ausführlich mit Hilfe von [Fig. 11A](#) bis [Fig. 11H](#) beschrieben. In diesem Fall sei angenommen, dass der Makroblock "2" durch einen Fehlerstartcode unterbrochen ist, wie in [Fig. 11A](#) gezeigt ist, und ein Bitstrom, in welchem Daten des Makroblocks nicht vollständig sind, zur variablen Längendecodierschaltung **18** geliefert wird.

[0109] In diesem Fall sei angenommen, dass der Makroblock "1" die gesamten Daten vom Block "1-1" bis zum Block "1-6" enthält und die Makroblockdaten vollständig sind. Andererseits ist im Fall des Makroblocks "2" der Block "2-2" nicht vollständig und durch den Fehlerstartcode D_{ES} unterbrochen.

[0110] Zunächst liefert die variable Längendecodierschaltung **18** den Makroblock "1", in welchem ein Bitstrom vorhanden ist, zum Ende. In diesem Fall ermöglicht die Verarbeitung des Makroblocks-Datenkopfs des Makroblocks "1", dass die variable Längendecodierschaltung **18** das DCT-Flag S8, das Quantisierungsmaß S4, den Vorhersagemodus S6, den Bewegungsvektor S5 und das Vorhersageflag S7 zur Schaltung einer hinteren Stufe liefert, und außerdem, dass die Bilddaten zur inversen Quantisierschaltung **19** geliefert werden, wenn jeder Block verarbeitet wird.

[0111] In Wirklichkeit jedoch kann die variable Längendecodierschaltung **18** nicht ermitteln, dass ein Bitstrom bis zum Ende für den Makroblock "1" vorhanden ist, bis die Verarbeitung aller Blöcke beendet ist, indem die Blöcke "1-1" bis "1-6" in der Reihenfolge verarbeitet werden, nachdem die Verarbeitung des Makroblock-Datenkopfs "1" endet. Daher wird nicht bestimmt, ob die Daten des Makroblocks "1" zur Schaltung der hinteren Stufe geliefert werden kön-

nen, bis die Verarbeitung aller Blöcke des Makroblocks "1" endet, so dass die variable Längendecodierschaltung **18** verschiedene Flags und Daten mit dem nachfolgenden zeitlichen Ablauf ausgibt.

[0112] Zunächst schreibt, wie in [Fig. 11C](#) gezeigt ist, die variable Längendecodierschaltung **18** das Quantisierungsmaß S4 und das DCT-Flag S8 aus dem DCT-Flag S8, dem Quantisierungsmaß S4, dem Vorhersagemodus S6, dem Bewegungsvektor S5 und dem Vorhersageflag S7 heraus in die inverse Quantisierungsschaltung **19** und die Rahmen-/Feld-DCT-Blockumordnungsschaltung **22**, wenn die Verarbeitung des Makroblock-Datenkopfs endet.

[0113] Danach beginnt die variable Längendecodierschaltung **18** mit der Verarbeitung in jedem Block, um das decodierte Bild jedes Blocks zu erlangen.

[0114] Da in diesem Fall eine Speicherschaltung für einen Makroblock nicht an der hinteren Stufe der variablen Längendecodierschaltung **18** vorgesehen ist, werden die decodierten Bilddaten unmittelbar zur inversen Quantisierungsschaltung **19** geliefert. Da jedoch die inverse Quantisierungsschaltung **19** und die IDCT-Schaltung **21** an der hinteren Stufe unter Annahme eines Blocks (8 Pixel × 8 Zeilen) als eine Einheit arbeiten, ist es auch möglich, die Bilddaten, welche durch die variable Längendecodierschaltung **18** verarbeitet wurden, über eine Speicherschaltung für einen Block zur inversen Quantisierungsschaltung **19** zu liefern.

[0115] Die inverse Quantisierungsschaltung **19**, die IDCT-Schaltung **21**, die Rahmen-/Feld-DCT-Blockumordnungsschaltung **22** führen jede Verarbeitung durch, wenn die Bilddaten jedes Blocks geliefert werden, wie in [Fig. 11D](#) gezeigt ist.

[0116] Wenn die gesamte Verarbeitung des Makroblocks "1" endet, gibt die variable Längendecodierschaltung **18** den Vorhersagemodus S6, den Bewegungsvektor S5, das Vorhersageflag S7 und das Maskierungsflag S9, die alle durch den Makroblock-Datenkopf erlangt werden, an den Bewegungskompensations-Schaltungsabschnitt **20** aus.

[0117] Da in diesem Fall die Verarbeitung durch die variable Längendecodierschaltung **18** für den Makroblock "1" beendet ist, werden der Vorhersagemodus S6, der Bewegungsvektor S5 und das Vorhersageflag S7 ausgegeben, die den Inhalt des Makroblock-Datenkopfs "1" haben, wie dieser ist.

[0118] In diesem Zusammenhang wird, da der Makroblock "1" ein vollständiger Makroblock ist, das Maskierungsflag S9, welches von der variablen Längendecodierschaltung **18** ausgegeben wird, als "0" bestimmt (d.h., keine Maskierung). Das Maskie-

rungsflag S9 wird mit dem Zeitablauf erzeugt, welcher in [Fig. 11E](#) gezeigt ist. Die oben erläuterten Prozesse sind die Verarbeitungen in den Schritten SP1 bis SP10.

[0119] Der Bewegungskompensationsschaltungsabschnitt **20** führt Decodierung gemäß dieser Flags durch.

[0120] Die Berechnungseinheit **23** empfängt die Bilddaten über die inverse Quantisierungsschaltung **19**, die IDCT-Schaltung **21** und die Rahmen-/Feld-DCT-Blockumordnungsschaltung **22** mit dem zeitlichen Ablauf, der in [Fig. 11F](#) gezeigt ist. In diesem Fall ist die Zeitdifferenz, welche zwischen der Zuführung der Flags und der Zuführung von Daten zur Berechnungseinheit **23** vorhanden ist, eine Verzögerungszeit, welche für die Verarbeitung durch die Bewegungskompensationsschaltung **24** erforderlich ist.

[0121] Danach addiert die Berechnungseinheit **23** die Bilddaten **510**, welche von einem Bitstrom erlangt werden, und das Vorhersagebild S11, welches von der Bewegungskompensationsschaltung **24** gelesen wird, mit dem Zeitablauf, der in [Fig. 11F](#) und [Fig. 11G](#) gezeigt ist, und gibt das Additionsergebnis als reproduziertes Ausgabebild S12 aus. Da in diesem Fall die Bestimmung des Maskierungsflags S9 zeigt "Nichtvorhandensein einer Maske", werden beide Bilddaten S10 und S11 dem Makroblock "1" hinzugefügt.

[0122] Anschließend wird die Verarbeitung für den Makroblock "2" einschließlich des Fehlerstartcodes beschrieben.

[0123] Die variable Längendecodierschaltung **18** ermittelt eine nicht normale Begrenzung, welche in einem Bitstrom durch den Fehlerstartcode DES erzeugt wird, einschließlich eines Synchroncodes. In [Fig. 11B](#) und [Fig. 11C](#) ist ein Fall beschrieben, bei dem ein Fehlerstartcode D_{ES} in der Mitte der Verarbeitung des Blocks "2-2" gefunden wird.

[0124] Die variable Längendecodierschaltung **18** verarbeitet den Makroblock-Datenkopf "2" und den Block "2-1" ähnlich wie beim Fall des Makroblocks "1". Der Block "2-1" ist kein Pseudo-Datenwort.

[0125] Die variable Längendecodierschaltung **18** liefert das Quantisierungsmaß S4 und das DCT-Flag S8 zur Schaltung der hinteren Stufe mit dem Zeittakt, der in [Fig. 11C](#) gezeigt ist, wenn die Verarbeitung des Makroblock-Datenkopfs "2" ähnlich dem Fall des Makroblocks "1" endet. Danach verarbeitet die variable Längendecodierschaltung **18** den Block "2-1" und liefert die Bilddaten für den Block "2-1" zur inversen Quantisierungsschaltung **19**.

[0126] Die variable Längendecodierschaltung **18**

findet den Fehlerstartcode, während der Block "2-2" verarbeitet wird. In diesem Zeitpunkt erzeugt die variable Längendecodierschaltung **18** Pseudodaten für den verbleibenden Teil des Makroblocks "2", während ein Bitstrom ausfällt, um so den Makroblock "2" zu vervollständigen, der in Betracht gezogen wird, dass die inverse Quantisierungsschaltung **19** und die IDCT-Schaltung **21** Verarbeitung in Blöcken durchführen und die Bewegungskompensationsschaltung **24** Verarbeitung in Makroblöcken durchführt. Dieser Zustand ist in [Fig. 11B](#) gezeigt.

[0127] In diesem Fall werden die Pseudodaten, welche aus einem Wert bestehen, beispielsweise null, an die inverse Quantisierungsschaltung **19** als die verbleibenden Bilddaten des Blocks "2-2", in welchem der Fehlerstartcode gefunden ist, und die Bilddaten der verbleibenden Blöcke "2-3", "2-4", "2-5", und "2-6" ausgegeben. Da in diesem Zusammenhang die Bilddaten des Makroblocks "2" durch das Maskierungsflag S9 bestimmt sind, so dass sie nicht für die Betätigung in der Berechnungseinheit **23** verwendet werden, ist es möglich, irgendeinen Wert mit Ausnahme von null als Wert der Pseudodaten zu verwenden.

[0128] Die inverse Quantisierungsschaltung **19**, die IDCT-Schaltung **21** und die Rahmen-/Feld-DCT-Blockumordnungsschaltung **22** führen jede Verarbeitung gemäß den Bilddaten aus, die aus den Pseudodaten zusammengesetzt sind, und liefern die Verarbeitungsergebnisse zur Berechnungseinheit **23**.

[0129] Der Vorhersagemodus S6, der Bewegungsvektor S5, das Vorhersageflag S7 und das Maskierungsflag S9, welche in der Bewegungskompensationsschaltung **24** verwendet werden, werden in dem Zeitpunkt ausgegeben, wenn die gesamte Verarbeitung des Makroblocks "2" endet. In diesem Fall ist die gesamte Verarbeitung als Erzeugung von Pseudodaten bis zum Block "2-6" definiert.

[0130] Da wie oben beschrieben der Makroblock "2" durch den Fehlerstartcode D_{ES} unterbrochen wird und dadurch nicht nützliche Bilddaten verarbeitet werden, ist die Steuerung so, dass die nicht nützlichen Bilddaten die Wiedergabe aufgrund des Kopierens eines Vorhersagebilds nicht beeinträchtigen. Das Kopieren des Vorhersagebilds ist eine Verarbeitung, welche mittels des Vorhersagemodus S6, des Bewegungsvektors S5, des Vorhersageflags S7 und des Maskierungsflags S8 und außerdem unter Verwendung der Bewegungskompensationsschaltung **24** und der Berechnungseinheit **23** durchgeführt wird.

[0131] Das erzeugte Vorhersagebild S11 weist das Vorhersagebild an der räumlich gleichen Position wie das vordere Bild auf, wie im vorherigen Abschnitt beschrieben (Schritte SP32 bis SP33), das Vorhersagebild an der räumlich gleichen Position wie einen Rah-

men vor dem Anzeigezeitpunkt (Schritte SP35 bis SP37), das Vorhersagebild unmittelbar vor einem Makroblock, der durch den Fehlerstartcode D_{ES} unterbrochen wurde und mittels des Vorhersagemodus S6 erlangt wurde, des Bewegungsvektors S5 und des Vorhersageflags S7 eines Makroblocks, der vorgeschlagen wurde (Schritte SP38 bis SP39), und das Vorhersagebild, welches unter Verwendung des Vorhersagemodus S6, des Bewegungsvektors S5 und des Vorhersageflags S7, welches durch einen verarbeiteten Makroblock-Datenkopf bestimmt wurde (Schritte SP40 bis SP42), erlangt wurde.

[0132] Die variable Längendecodierschaltung **18** erlangt den Vorhersagemodus S6, den Bewegungsvektor S5 und das Vorhersageflag S7 zum Erlangen dieser Vorhersagebilder SV und gibt diese an die Bewegungsermittlungsschaltung **24** aus. Außerdem zeigt in diesem Fall die Bestimmung des Maskierungsflags S9 "Vorhandensein der Maskierung" (d.h., "1").

[0133] In diesem Zusammenhang wird, wenn der Makroblock "2" lediglich aus den Daten im aktuellen Bild ohne Verwendung von Vorhersage zusammengesetzt ist, die Prozedur geändert, um ein Vorhersagebild mittels des Vorhersagemodus S6, des Bewegungsvektors S5 und des Vorhersageflags S7 des benachbarten Makroblocks zu erzeugen.

[0134] Die Bewegungskompensationsschaltung **24** erzeugt ein Vorhersagebild gemäß den Flags, wie in [Fig. 11G](#) gezeigt ist. Außerdem fügt die Berechnungseinheit **23** die Pseudobilddaten S10, welche über die inverse Quantisierungsschaltung **19** oder dgl. zugeführt werden, und das Vorhersagebild S11, welches von der Bewegungskompensationsschaltung **24** zugeführt wird, hinzu und gibt diese aus. Da jedoch "Vorhandensein der Maske" durch das Maskierungsflag S9 gezeigt wird, werden die Bilddaten S10 für die Berechnung nicht verwendet, sondern das Vorhersagebild S11 wird unmittelbar ausgegeben und dient als reproduziertes Ausgangsbild S12.

[0135] Gemäß dem obigen Aufbau ist es möglich, ein Bilddaten-Decodiergerät zu realisieren, welches in der Lage ist, einen nichtvollständigen Makroblock zu betreiben, der durch den Fehlerstartcode D_{ES} beschränkt ist, ohne den Betrieb der internen Schaltungen der Decodierschaltung **16** zu beeinträchtigen, welche blockweise oder in Makroblöcken arbeiten. Außerdem erfordert diese Ausführungsform keine spezielle Zeittakt-Steuerschaltung, um die internen Schaltungen korrekt zu betreiben. Daher ist es nicht notwendig, sich um die Vergrößerung des Schaltungsaufbaus zu kümmern.

[0136] Da jedoch ein Wiedergagebild nicht durch Ersetzen eines Vorhersagebilds für das Bilddatenteil eines nichtvollständigen Makroblocks beeinträchtigt

wird, um den Makroblock auszugeben, ist es möglich, eine Bilddaten-Decodiervorrichtung zu realisieren, die es ermöglicht, ein Bild mit einer überragenden Qualität im Vergleich zu einem herkömmlichen Bild zu erlangen.

(3) Weitere Ausführungsformen

[0137] Bei der obigen Ausführungsform verwendet der Fehlerstartcode D_{ES} einen 32-Bit-Code, ausgedrückt durch "00000000 00000000 00000000 00000001 10110100". Die vorliegende Erfindung ist jedoch nicht darauf beschränkt, sondern es kann irgendein Code einschließlich eines Synchroncodes verwendet werden. Außerdem ist die Codelänge nicht auf 32 Bits beschränkt, sondern es kann ein Code mit einer Länge von mehr als 32 Bits verwendet werden.

[0138] Außerdem wird bei der obigen Ausführungsform vermieden, dass die Arbeitsweise der Decodierschaltung **16** beeinträchtigt wird, indem der Fehlerstartcode D_{ES} in einen Bitstrom eingefügt wird. Die vorliegende Erfindung ist jedoch nicht darauf beschränkt, sondern es kann der Fehlerstartcode D_{ES} für weitere Verarbeitung, beispielsweise die Spezialreproduktion verwendet werden.

[0139] Außerdem werden bei der obigen Ausführungsform transformierte Daten gemäß diskreter Kosinus-Transformation im Zeitpunkt der Decodierung verwendet. Die vorliegende Erfindung ist jedoch nicht darauf beschränkt, sondern kann breit angewandt werden, orthogonal-transformierte Daten zu decodieren.

[0140] Außerdem wird bei der obigen Ausführungsform ein Videosignal, welches über einen Aufzeichnungsträger **3** oder eine Übertragungsleitung zugeführt wird, decodiert. Die vorliegende Erfindung ist jedoch nicht darauf beschränkt, sondern sie kann auch dazu angewandt werden, ein Hörsignal oder ein Steuersignal zusätzlich zum Videosignal zu decodieren.

[0141] Wie oben beschrieben wird ein nichtkorrigierbares Fehlerteil eines Digitalsignals oder eines Datenteils einschließlich des nichtkorrigierbaren Fehlerteils durch einen Spezialcode ersetzt, der einen Synchroncodes aufweist, und an die Decodiereinrichtung ausgegeben, so dass es möglich ist, eine Digitalsignal-Decodiervorrichtung zu erlangen, welche das Vorhandensein eines Fehlers im Decodierzeitpunkt bestätigen und genaue Fehlerverarbeitung ausführen kann.

[0142] Wenn außerdem eine Datengruppe einer vorher festgelegten Länge, welche ein Digitalsignal bildet, durch einen vorher festgelegten Code begrenzt ist und nicht vollständig ist, gibt die Decodier-

einrichtung ein digitales Vorhersagesignal anstelle eines decodierten Signals der nichtvollständigen Datengruppe aus. Daher ist es möglich, eine Digitalsignal-Decodiervorrichtung zu erlangen, welche ein Ausgangssignal erlangen kann, welches nicht durch die Datengruppe beeinträchtigt wird, ohne eine Spezialschaltung hinzuzufügen, um die Datengruppe zu verarbeiten.

[0143] Obwohl in diesem Zusammenhang bevorzugte Ausführungsformen der Erfindung beschrieben wurden, soll es für den Fachmann klar sein, dass verschiedene Änderungen und Modifikationen durchgeführt werden können, so dass somit die beigefügten Patentansprüche alle derartigen Änderungen und Modifikationen abdecken sollen, die innerhalb des Rahmens der Erfindung fallen.

Patentansprüche

1. Digitalsignal-Decodiervorrichtung (**12**), um ein codiertes digitales Videosignal, welches Makroblöcke aufweist, über einen Übertragungsweg zu empfangen, um ein Ausgangssignal von einem decodierten Signal des codierten digitalen Videosignals und eines vorhersagten digitalen Videosignals zu erzeugen, welche aufweist:

eine Decodierverarbeitungseinrichtung (**13**) zum Ausgeben des vorhergesagten digitalen Videosignals anstelle des decodierten Signals für den gesamten Makroblock des Digitalsignals, wenn der Makroblock durch einen vorher bestimmten Fehlerstartcode abgegrenzt ist, der das Vorhandensein eines Fehlers im Digitalsignal zeigt, und dadurch unvollständig ist.

2. Digitalsignal-Decodiervorrichtung nach Anspruch 1, wobei die Decodierverarbeitungseinrichtung (**13**) eine Variabel-Längen-Decodierschaltung (**18**) hat, und die Variabel-Längen-Decodierschaltung (**18**) verschiedene Flag-Signale, die verwendet werden, das Vorhersage-Digitalsignal zu erzeugen, zu einer Schaltung einer hinteren Stufe überträgt, wenn alle Verarbeitungen für den Makroblock beendet sind.

3. Digitalsignal-Decodiervorrichtung nach Anspruch 2, wobei ein Ausgabeverbotsflag zum Verbiehen der Verwendung des decodierten Signals des Digitalsignals in den verschiedenen Flag-Signalen enthalten ist.

4. Digitalsignal-Decodiervorrichtung nach Anspruch 3, wobei die Digitalsignal-Decodierschaltung (**18**) Pseudodaten in einen mangelhaften Teil des unvollständigen Makroblocks einfügt, so dass die Verarbeitung in einer Schaltung einer hinteren Stufe beendet wird, wenn der nichtvollständige Makroblock zugeführt wird.

5. Digitalsignal-Decodiervorrichtung nach Anspruch 3, wobei:

die Schaltung, welche in der hinteren Stufe der Variabel-Längen-Decodierschaltung (18) angeordnet ist, eine Berechnungsschaltung hat, um ein decodiertes Signal des Digitalsignals und des Vorhersage-Digitalsignals zu berechnen, um somit ein Ausgabesignal zu erzeugen, und die Berechnungsschaltung den Wert des decodierten Signals entsprechend dem Makroblock als null betrachtet, wenn die Verwendung des decodierten Signals durch das Ausgabeverbotsflag verboten wird.

6. Digitalsignal-Decodiervorrichtung nach Anspruch 3, wobei:

die Schaltung, welche in der hinteren Stufe der Variabel-Längen-Decodierschaltung (18) angeordnet ist, eine Berechnungsschaltung hat, um ein decodiertes Signal des Digitalsignals und des Vorhersage-Digitalsignals zu berechnen, um somit ein Ausgabesignal zu erzeugen, und die Berechnungsschaltung das decodierte Signal entsprechend dem Makroblock für den Betrieb nicht verwendet, wenn die Verwendung des decodierten Signals durch das Ausgabeverbotsflag verboten wird.

7. Digitalsignal-Decodiervorrichtung nach Anspruch 1, wobei:

das Vorhersage-Digitalsignal ein Bild an der räumlich gleichen Position wie ein Voraus-Bild ist.

8. Digitalsignal-Decodiervorrichtung nach Anspruch 1, wobei:

das Vorhersage-Digitalsignal ein Bild an der räumlich gleichen Position wie ein Vorhersage-Bild einen Rahmen vor dem Anzeigezeitpunkt ist.

9. Digitalsignal-Decodiervorrichtung nach Anspruch 1, wobei:

das Vorhersage-Digitalsignal ein Bild unmittelbar vor der unvollständigen Datengruppe ist, die durch den vorher festgelegten Code abgegrenzt ist, und mittels verschiedener Flag-Signale eines vorhergesagten Makroblocks erhalten wird.

10. Digitalsignal-Decodiervorrichtung nach Anspruch 1, wobei:

das vorhergesagte Digitalsignal ein Bild ist, welches mittels verschiedener Flag-Signale erhalten wird, die durch einen Datenkopf bestimmt werden, wenn der Prozess des Datenkopfteils des Makroblocks beendet ist.

Es folgen 11 Blatt Zeichnungen

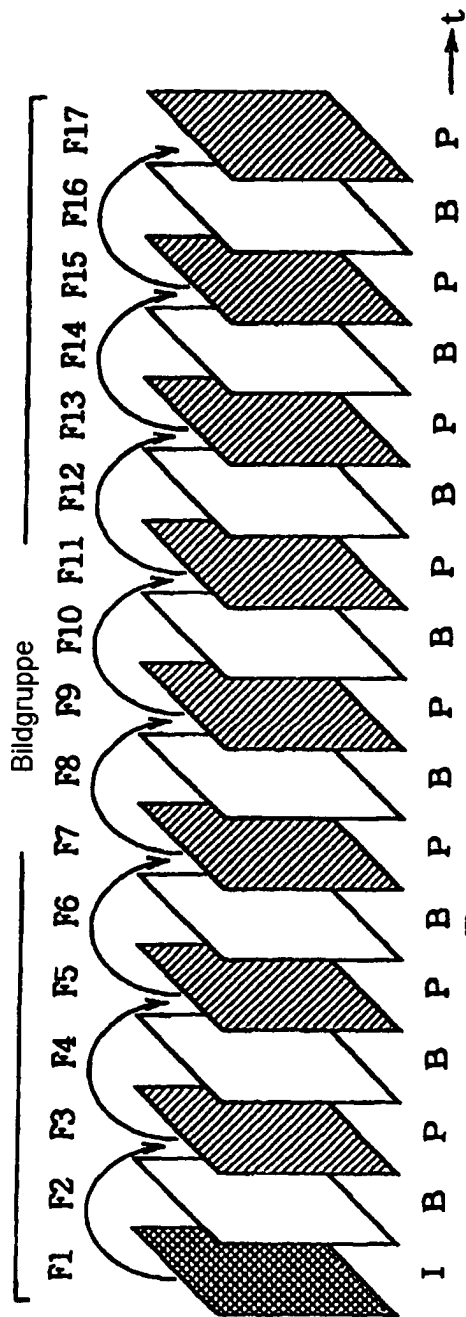


FIG. 1A (Stand der Technik)

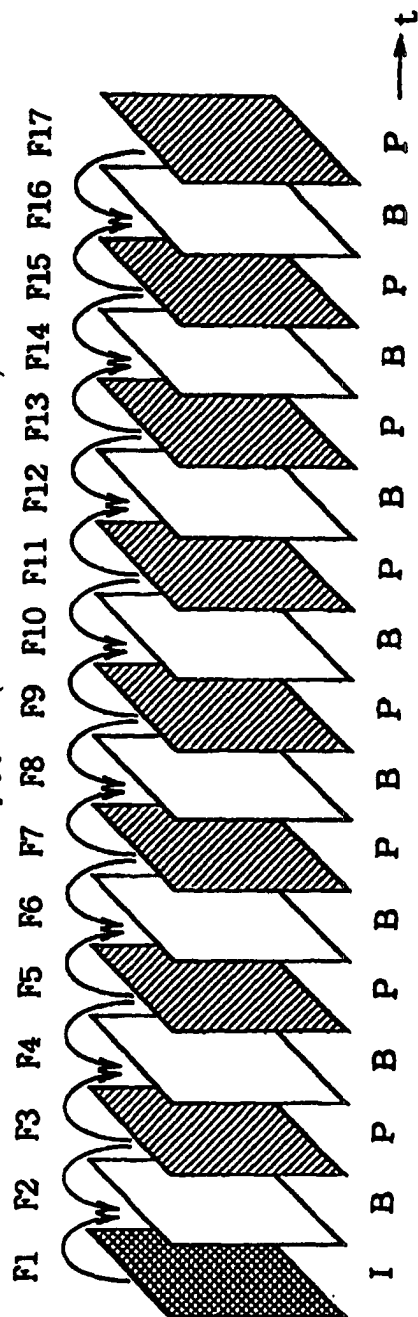


FIG. 1B (Stand der Technik)

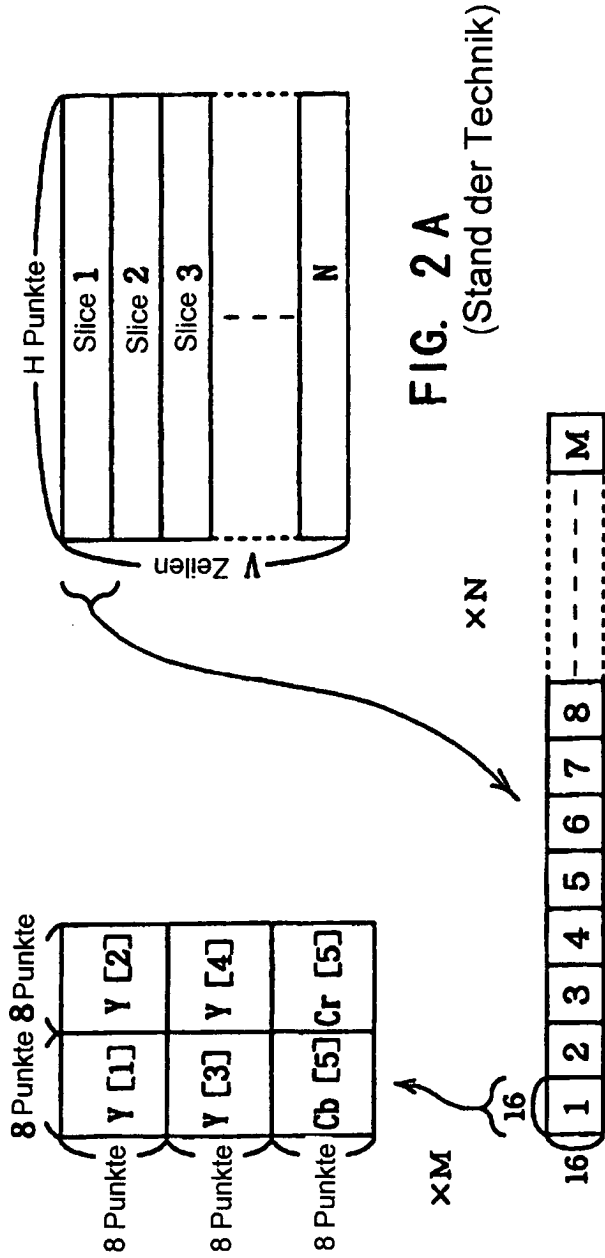


FIG. 2 C
(Stand der Technik)

FIG. 2 A
(Stand der Technik)

FIG. 2 B
(Stand der Technik)

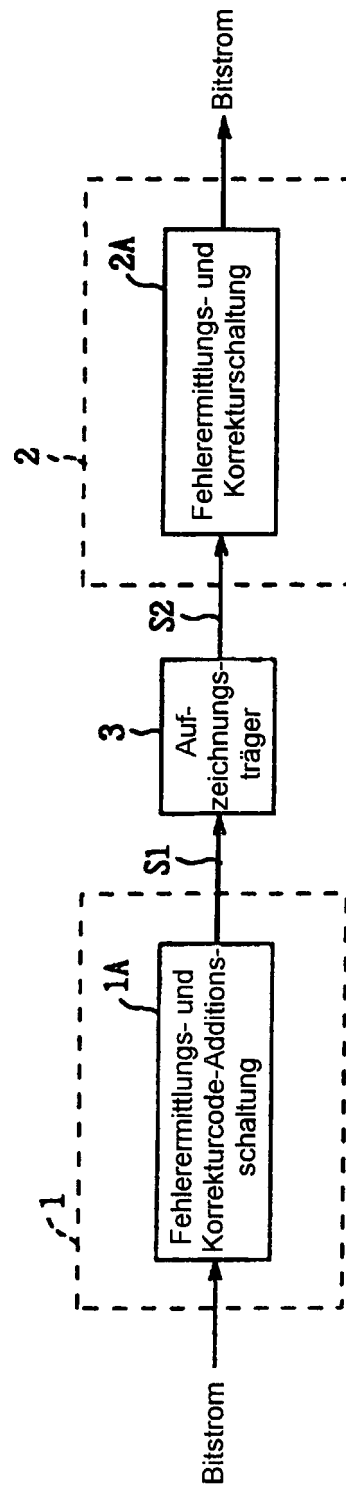
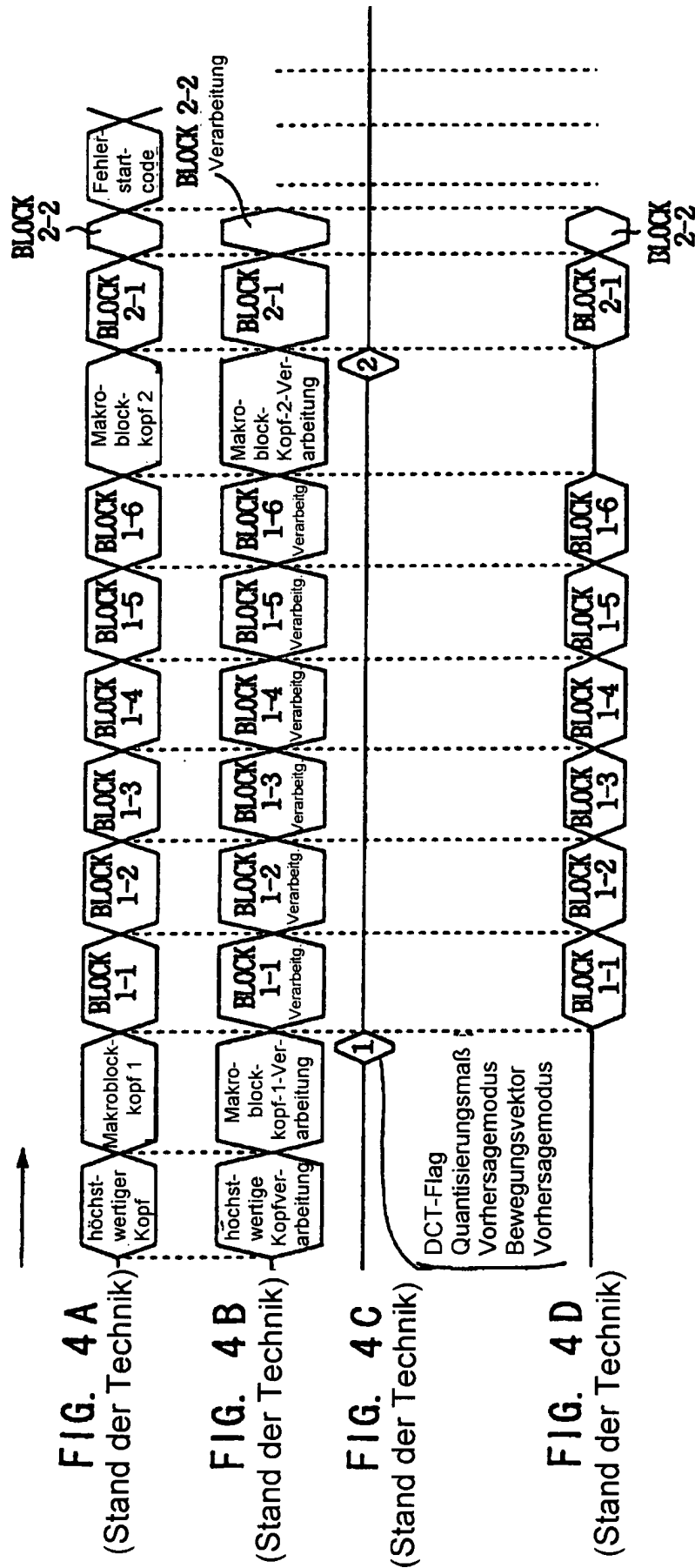


FIG. 3



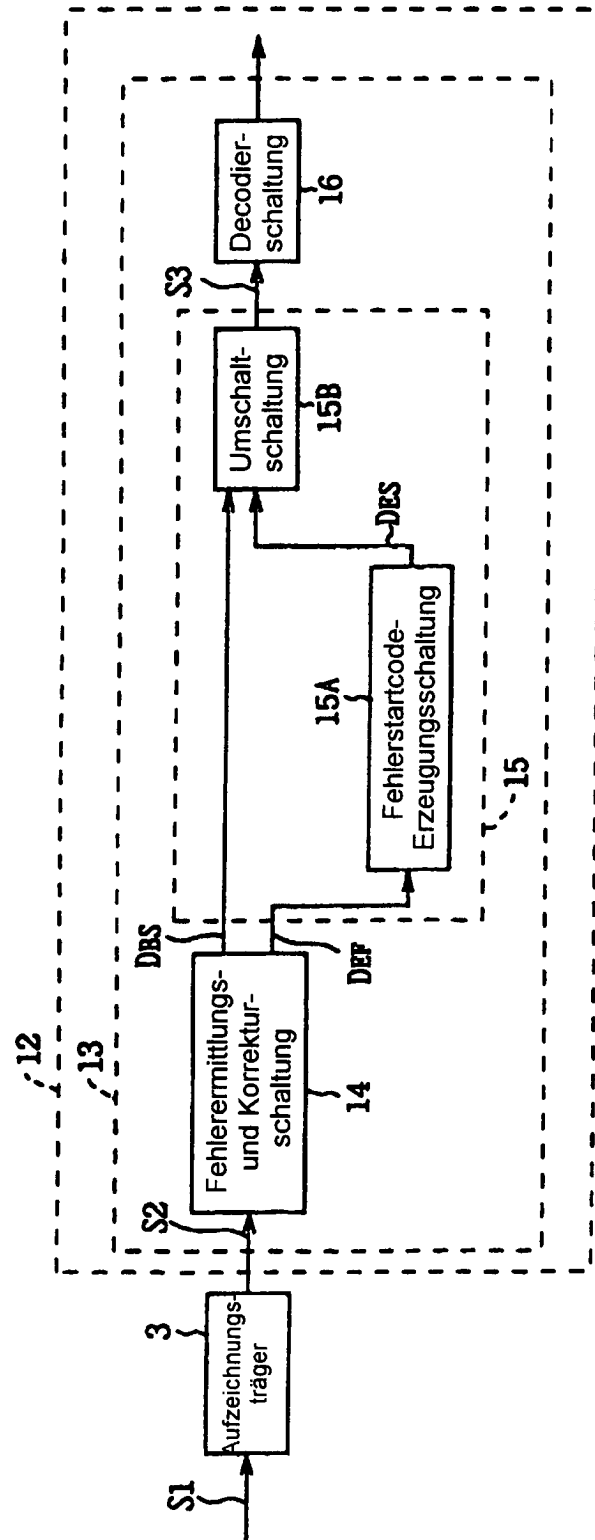


FIG. 5

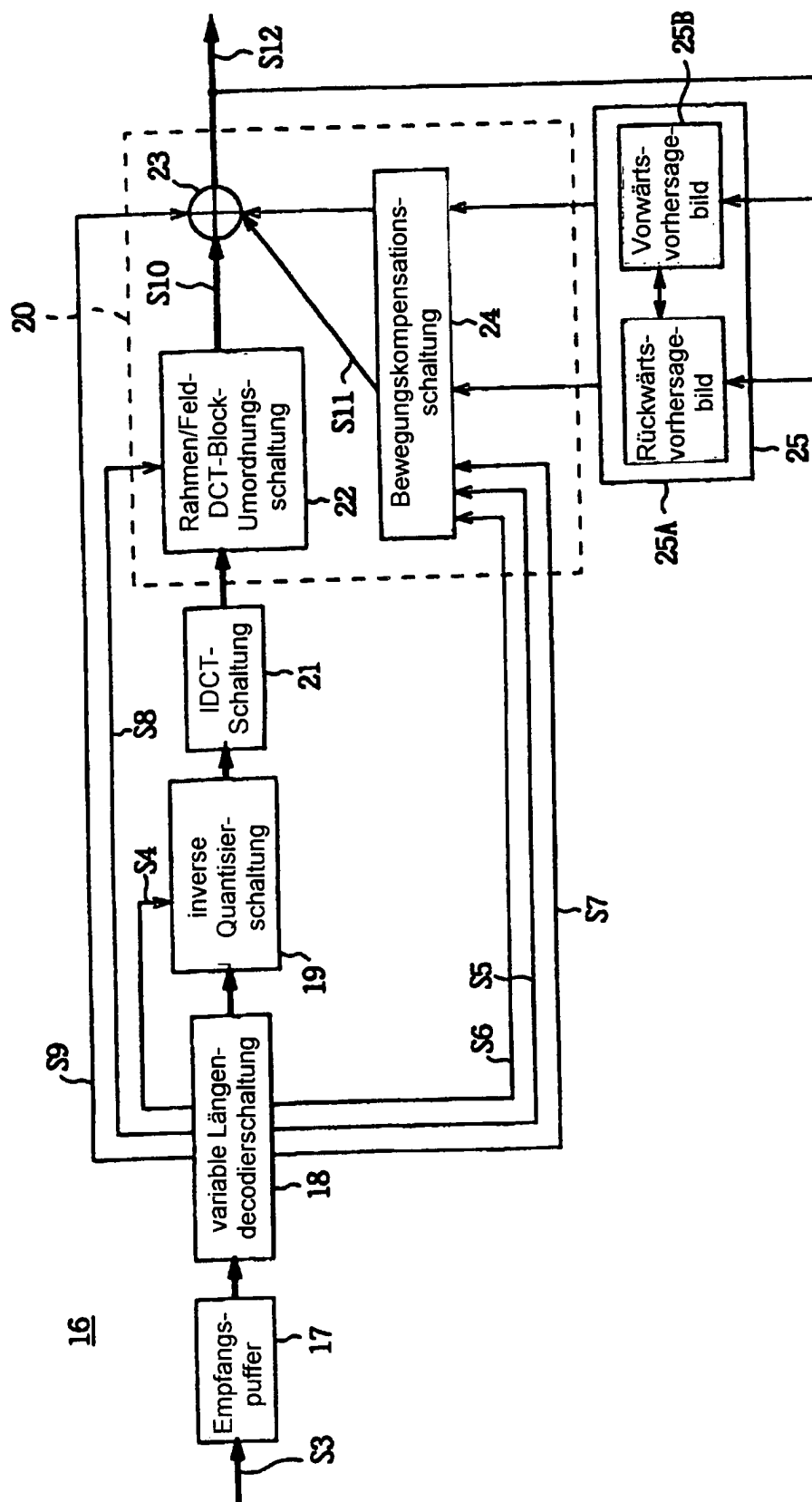


FIG. 6

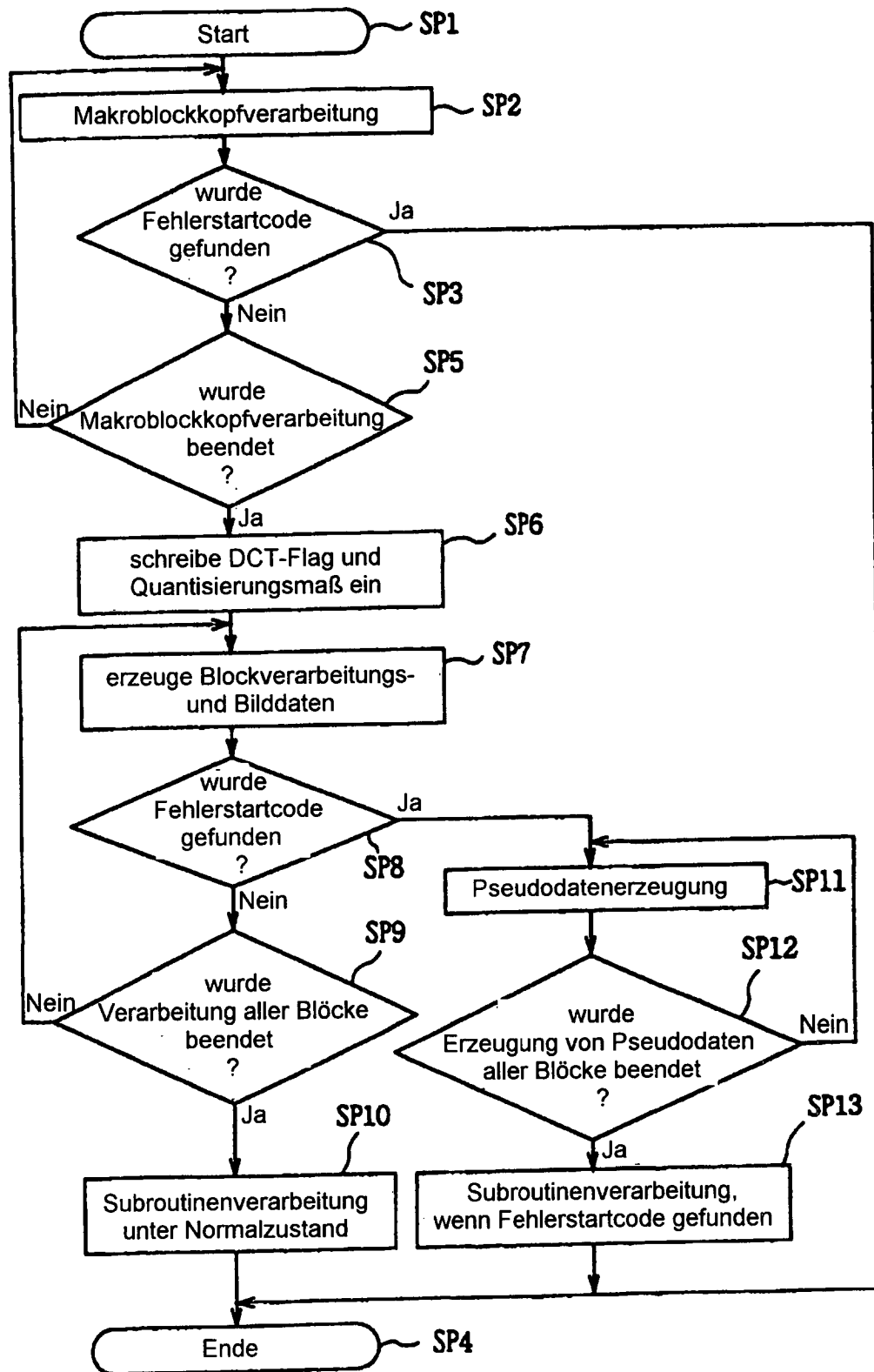


FIG. 7

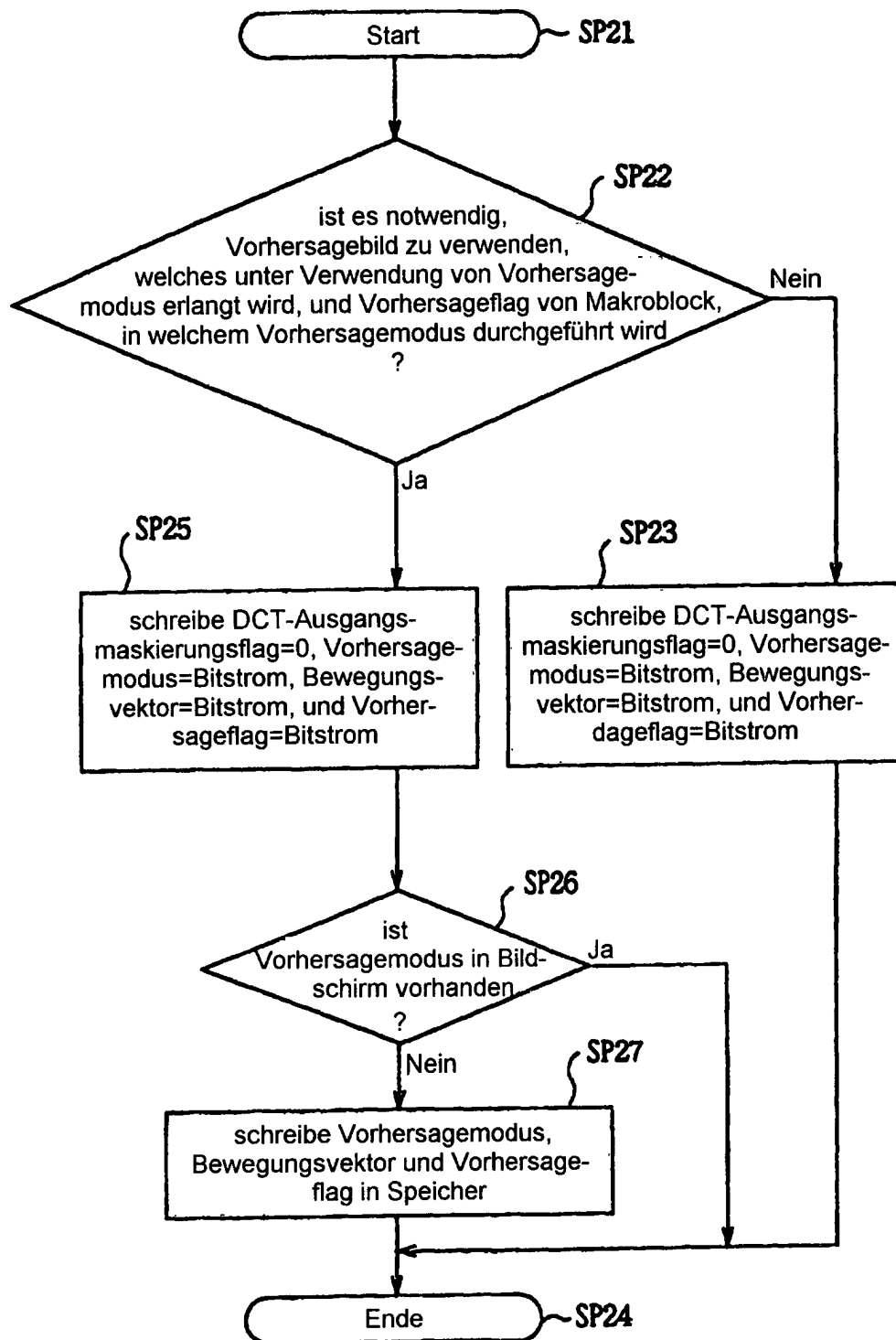
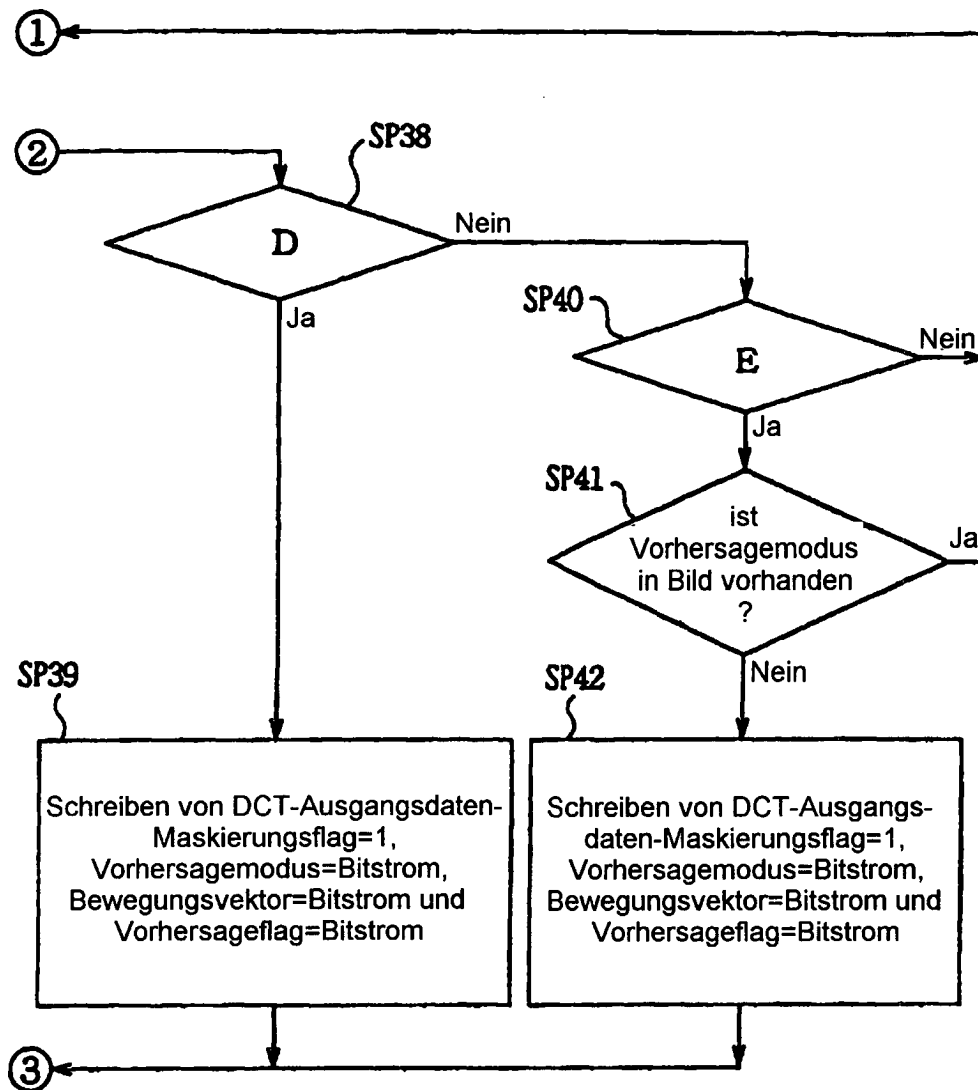


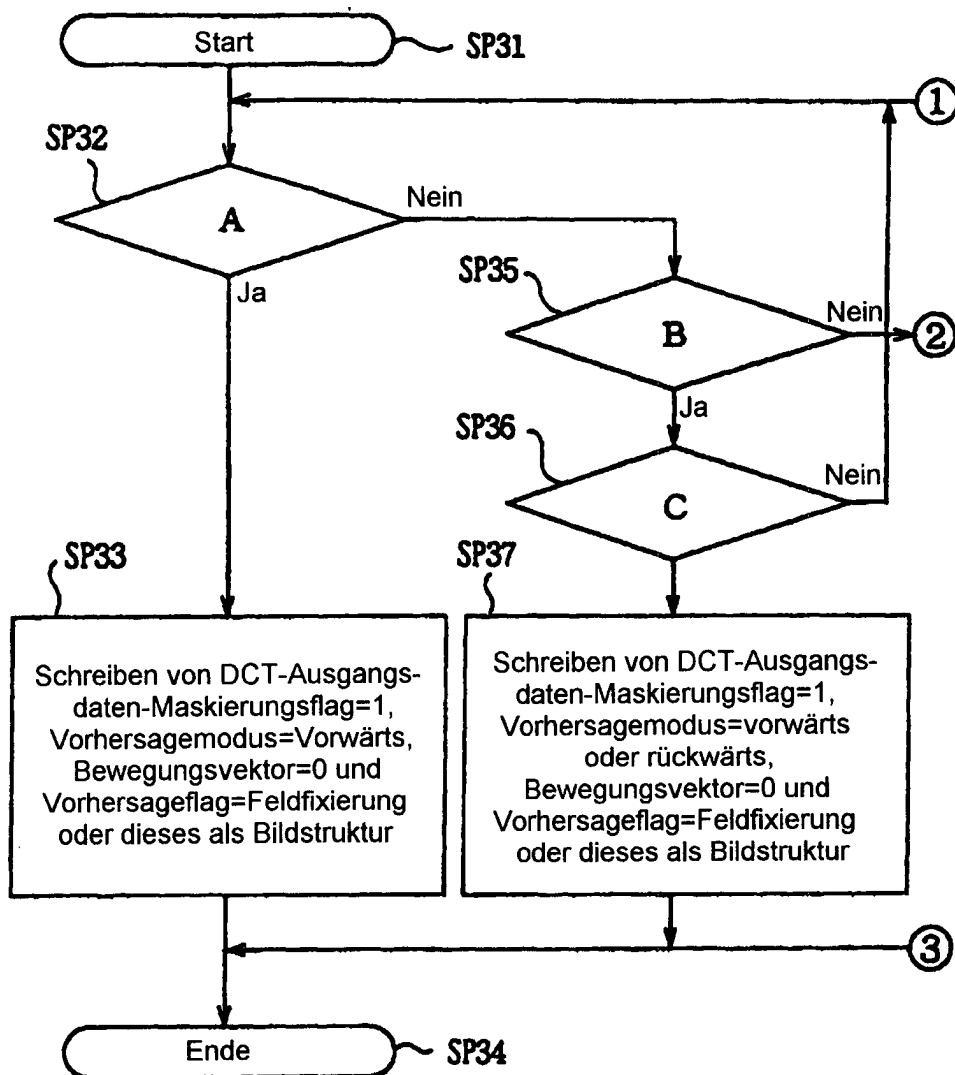
FIG. 8



D: ist es notwendig, Vorhersagebild zu verwenden, welches unter Verwendung von Vorhersagemodus, Bewegungsvektor und Makroblock-Vorhersageflag erlangt wird, bei dem vorherige Vorhersage durchgeführt wird ?

E: es ist notwendig, Vorhersagebild zu verwenden, welches unter Verwendung von Vorhersagemodus, Bewegungsvektor und Vorhersageflag von verarbeiteten Makroblock erlangt wurde ?

FIG. 9



- A: ist es notwendig, Vorhersagebild bei räumlich gleicher Position wie vorderes Bild zu verwenden ?
- B: ist es notwendig, Vorhersagebild bei räumlich gleicher Position wie Bild einen Rahmen vorher in Anzeigezeit zu verwenden ?
- C: ist es I-Bild, P-Bild oder B-Bild, deren unmittelbares vorhergehendes nicht B-Bild ist ?

FIG. 10

