



(19)
Bundesrepublik Deutschland
Deutsches Patent- und Markenamt

(10) **DE 697 32 076 T2 2005.12.08**

(12)

Übersetzung der europäischen Patentschrift

(97) **EP 0 836 285 B1**

(21) Deutsches Aktenzeichen: **697 32 076.6**

(96) Europäisches Aktenzeichen: **97 307 852.0**

(96) Europäischer Anmeldetag: **03.10.1997**

(97) Erstveröffentlichung durch das EPA: **15.04.1998**

(97) Veröffentlichungstag

der Patenterteilung beim EPA: **29.12.2004**

(47) Veröffentlichungstag im Patentblatt: **08.12.2005**

(51) Int Cl.7: **H03M 13/15**

(30) Unionspriorität:

26711796 08.10.1996 JP

(73) Patentinhaber:

Canon K.K., Tokio/Tokyo, JP

(74) Vertreter:

TBK-Patent, 80336 München

(84) Benannte Vertragsstaaten:

DE, FR, GB, IT, NL

(72) Erfinder:

Shimizu, Tetsuya, Ohta-ku, Tokyo, JP

(54) Bezeichnung: **Reed-Solomon Dekodierer mit universeller Prozessoreinheit und speziellen Schaltungen**

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 eine Fehlerkorrekturdekodiervorrichtung zum Korrigieren eines Codefehlers in gesendeten Codedaten.

Verwandter Stand der Technik

[0002] Seit kurzem erhöht sich die Dichte einer digitalen Aufzeichnung ständig, und dementsprechend wird eine Fehlerkorrekturkodier/-dekodiervorrichtung gefordert, die mit hoher Geschwindigkeit arbeiten kann und eine hohe Korrekturleistung hat.

[0003] Ein Reed-Solomon-Code ist eine Art BCH-Code und weist eine hohe Korrekturleistung auf, da er in Byteeinheiten korrigiert werden kann. Dementsprechend wird der Reed-Solomon-Code auf einer Vielzahl von Gebieten wie optischen Platten und digitaler Videoaufzeichnung extensiv verwendet.

[0004] Nachstehend wird das Prinzip eines repräsentativen Dekodierverfahrens für einen Reed-Solomon-Code kurz beschrieben.

[0005] Ein q -Element-Reed-Solomon-Code mit α als primitivem Element von $GF(q)$ und $1, \alpha, \alpha^2, \dots, \alpha^{d-2}$ als Wurzel eines Generatorpolynoms $G(x)$ wird als Beispiel verwendet. Die Codelänge ist $n = q - 1$, die Informationssymbolanzahl ist $k = q - d$ und die minimale Distanz ist $d_{\min} = d$.

[0006] Zuerst wird ein Syndrom S_i aus einem empfangenen Polynom $Y(x)$ anhand der nachstehenden Gleichung (1) berechnet.

$$S_i = Y(\alpha^i) = E(\alpha^i) \quad (i = 0, 1, \dots, d - 2) \quad (1)$$

wobei $E(x)$ ein Fehlermuster darstellendes Polynom ist.

[0007] Es wird angenommen, dass ein Fehler an einer Position j_1, j_2, \dots, j_l auftritt. Es gilt

$$n - 1 \geq j_1 > j_2 > \dots > j_l \geq 0 \quad (2)$$

$$1 \leq t = \lceil (d - 1)/2 \rceil \quad (3)$$

wobei $\lceil x \rceil$ eine maximale ganze Zahl kleiner als x ist, und t die Fehlerkorrekturleistung angibt.

[0008] Dann ist ein Fehlermusterpolynom $E(x)$ wie folgt dargestellt

$$E(x) = E_1 x^{j_1} + E_2 x^{j_2} + \dots + E_l x^{j_l} \quad (4)$$

E_i ($i = 1, \dots, l$) ist der Wert eines Fehlers an einer Position j_i und des Elements von $GF(q)$. Das Syndrom S_i wird anhand Gleichung (1) wie folgt geschrieben.

$$S_i = E_1 x_1^i + E_2 x_2^i + \dots + E_l x_l^i \quad (5)$$

$$\text{für } x_i = \alpha^{j_i} \quad (i = 1, \dots, l) \quad (6)$$

[0009] Ein Reed-Solomon-Code wird durch Berechnung der Fehlerpositionen j_1, j_2, \dots, j_l und der Fehlerwerte E_1, E_2, \dots, E_l aus dem Syndrom S_i dekodiert. Zu diesem Zweck werden zwei Polynome

$$\sigma(z) = \sigma_l z^l + \dots + \sigma_1 z + \sigma_0 \quad (7)$$

$$\begin{aligned}\eta(z) &= \eta_{l-1}z^{l-1} + \dots + \eta_1z + \eta_0 & (8) \\ &= \sum_{i=1}^l E_i \prod_{k=i}^l (1 - \alpha_k z)\end{aligned}$$

über $GF(q)$ berechnet. Diese Polynome haben $\alpha^{-j_1}, \alpha^{-j_2}, \dots, \alpha^{-j_l}$ als Wurzel. $\sigma(z)$ wird Fehlerpositionspolynom genannt, und $\eta(z)$ ist ein Polynom, das zur Berechnung des Fehlerwerts erforderlich ist, und wird Fehlerauswertungspolynom genannt.

[0010] Folgendes Polynom mit dem Syndrom als Koeffizienten wird betrachtet.

$$S(z) = S_{d-2}z^{d-2} + \dots + S_1z + S_0 \quad (9)$$

[0011] Die Beziehung zwischen dem Syndrompolynom, $\sigma(z)$ und $\eta(z)$ kann wie folgt geschrieben werden.

$$\sigma(z)S(z) = \eta(z) \text{ mod } z^{d-1} \quad (10)$$

[0012] Dies ist eine grundlegende Gleichung, aus der $\sigma(z)$ und $\eta(z)$ berechnet werden können.

[0013] Ist die Fehlerkorrekturleistung t 4 bis 5, kann ein Verfahren beruhend auf einem Euklidischen Verfahren wechselseitiger Teilung als Verfahren zum effektiven Berechnen von $\sigma(z)$ und $\eta(z)$ verwendet werden. Dieses Verfahren wird nachstehend als Euklidischer Algorithmus bezeichnet. Gleichung (10) kann unter Verwendung eines geeigneten Polynoms $A(z)$ wie folgt umgeschrieben werden.

$$A(z)z^{d-1} + \sigma(z)S(z) = \eta(z) \quad (11)$$

[0014] Da angenommen wird, dass die Fehlerzahl t oder kleiner ist, muss folgende Gleichung erfüllt sein.

$$\deg \eta(z) < \deg \sigma(z) \leq t \quad (12)$$

wobei \deg der Grad eines Polynoms ist.

[0015] Wie es aus den Gleichungen (8) und (7) ersichtlich ist, haben $\sigma(z)$ und $\eta(z)$ keinen gemeinsamen Faktor. Es ist bewiesen, dass $\sigma(z)$ und $\eta(z)$, die die Gleichung (12) erfüllen, eindeutig bestimmt werden können. Es ist auch bekannt, dass $\sigma(z)$ und $\eta(z)$ durch den Euklidischen Algorithmus zum Berechnen des größten gemeinsamen Teilerpolynoms von z^{d-1} und $S(z)$ berechnet werden können. [Fig. 1](#) zeigt ein allgemeines Verfahren des euklidischen Algorithmus.

[0016] Als Verfahren zum Erhalten der Wurzel von $\sigma(z)$, die durch den Euklidischen Algorithmus berechnet wird, wird im Allgemeinen eine Chien-Suche verwendet. Die Chien-Suche ist ein Verfahren, durch das die Potenz α^i ($i = 0, 1, \dots, n-1$) von α sequenziell in $\sigma(z)$ eingesetzt wird, und überprüft wird, ob $\sigma(\alpha^i)$ gleich 0 ist.

[0017] Fehlerwerte e_1, e_2, \dots, e_l können wie folgt berechnet werden.

$$\begin{aligned}e_i &= -\eta(\alpha^{-j_i})/\sigma(\alpha^{-j_i}) \\ (i &= 1, \dots, l)\end{aligned} \quad (13)$$

wobei $\sigma(z)'$ eine hergeleitete Funktion von $\sigma(z)$ ist und durch formelles Ableiten in $\sigma(z)$ berechnet wird.

[0018] Herkömmlicherweise sind die einzelnen Schritte der vorstehenden Korrekturverarbeitung im Allgemeinen als Pipeline ausgebildet und werden durch Hardware gebildet.

[0019] [Fig. 2](#) zeigt ein Beispiel der Anordnung einer herkömmlichen Reed-Solomon-Codefehlerkorrekturdekodiervorrichtung. Gemäß [Fig. 2](#) werden Eingangsdaten von einem Eingangsanschluss **600**, die einen Fehlerkorrektur-kodierten Fehler enthalten, um eine Verarbeitungszeit (die nachstehend beschrieben ist) durch eine Datenverzögerungsschaltung **601** verzögert. Gleichzeitig berechnet eine Syndromoperationsschaltung **602** das Syndrom S_i entsprechend Gleichung (1). Dieses Syndrom S_i wird als durch in Gleichung (2) dargestelltes Syndrompolynom einer Galois-Feldpolynomoperationsschaltung **603** zugeführt.

[0020] Die Galois-Feldpolynomoperationsschaltung **603** berechnet das Fehlerpositionspolynom $\sigma(z)$ und das

Fehlerwertpolynom $\eta(z)$ unter Verwendung des in [Fig. 1](#) gezeigten Euklidischen Algorithmus. Bei diesem Euklidischen Algorithmus werden primär ein Quotientenpolynom und ein Restpolynom eines Galois-Feldpolynoms wiederholt berechnet. Das eine Vergleichsbedingung $\deg R(z) \leq t - 1$ in der Galois-Feldpolynomoperationsschaltung **603** erfüllende Fehlerpositionspolynom $\sigma(z)$ wird einer Chien-Suchschaltung **604** zugeführt.

[0021] Die Chien-Suchschaltung **604** ersetzt sequenziell Elemente eines Galois-Feldes $GF(2^m)$ und überprüft, ob der jeweils resultierende Wert gleich 0 ist, wodurch eine Wurzel berechnet wird. Die Chien-Suchschaltung **604** führt dann Fehlerpositionsinformationen i , den Wert von $\eta(z)$ und den Wert $\sigma(z)$, der durch formelle Ableitung der Fehlerposition berechnet wird, einer Fehlerwertberechnungsschaltung **605** zu. Die Fehlerwertberechnungsschaltung **605** berechnet den Fehlerwert e_i aus den Fehlerpositionsinformationen i , die durch die Chien-Suchschaltung **604** berechnet werden, und Gleichung (13).

[0022] Die so berechnete Fehlerposition und der Fehlerwert werden einer Korrekturschaltung **606** zugeführt. Die Korrekturschaltung **606** führt eine Korrekturverarbeitung durch Berechnung einer Exklusiv-Oder-Verknüpfung der Fehlerposition in der durch die Datenverzögerungsschaltung **601** verzögerten Datenfolge und des berechneten Fehlerwerts durch. Die Korrekturschaltung **606** gibt das Ergebnis aus einem Ausgangsanschluss **607** aus.

[0023] Bei der vorstehenden herkömmlichen Anordnung erhöht sich allerdings der Schaltungsumfang zwangsläufig, indem alle Blöcke durch Hardware ausgebildet werden, obwohl eine Hochgeschwindigkeitsverarbeitung bewirkt werden kann. Außerdem muss der Hardwareaufbau des Steuerblocks immer dann verändert werden, wenn der Algorithmus verändert wird, um beispielsweise eine komplizierte wiederholte Dekodierung mit einer Löschkorrektur produktmodulierter Daten auszuführen.

[0024] Aus der US-A-4,899,341 ist eine BCH-Fehlerkorrekturschaltung bekannt, bei der eine Vielzahl von Galois-Operationsschaltungen parallel betrieben werden.

[0025] Aus der US-A-5,535,225 ist ferner bekannt, drei parallele Galois-Feldprozessoren in einem BCH-Dekodierer bereit zu stellen, der verschiedene Werte einer Codewortlänge und Anzahl von Informationssymbolen verarbeiten kann.

[0026] Aus 8049e IEE Proceedings on E. Computers and digital techniques, 1. Mai 1990, Nr. 3, Band 137, Teil E, Seiten 197–201, Stevenage (GB), M. Hahn, ist ferner bekannt, einen universellen Fehlerkorrekturkodierer-Dekodierer bereitzustellen, der einen Prozessor, FIFO und eine Reihe universeller Registerschaltungen (URC) verwendet. Der Prozessor berechnet ein Fehlerortpolynom und überträgt es über den FIFO zu einer URC zur Fehlerortberechnung. Durch den Prozessor berechnete Fehlerort- und Fehlerwertpolynome werden dann zu einer anderen URC zur Berechnung von Fehlerwerten übertragen.

[0027] Unter den vorstehend beschriebenen Umständen besteht die Aufgabe der Erfindung darin, eine Fehlerkorrekturdekodiervorrichtung auszubilden, die eine Hochgeschwindigkeitsverarbeitung mit kleinem Schaltungsumfang durchführen kann und eine hohe Einsatzflexibilität aufweist, sowie ein Verfahren für die Vorrichtung auszubilden.

[0028] Ausgestaltungen der Erfindung sind in den beigefügten unabhängigen Patentansprüchen dargelegt.

[0029] Gemäß einem bevorzugten Ausführungsbeispiel der Erfindung ist zur Lösung der vorstehenden Aufgabe eine Fehlerkorrekturdekodiervorrichtung ausgestaltet, mit einer ersten Operationseinrichtung mit einer Galois-Feldpolynomoperationsschaltung zum Erhalten eines Galois-Feldpolynoms, einer zweiten Operationseinrichtung mit einer Galois-Feldoperationsschaltung zur Durchführung einer Operation unter Verwendung des Galois-Feldpolynoms und einer Programmerzeugungseinrichtung zur Erzeugung eines Steuerprogramms für die erste und die zweite Operationseinrichtung, wobei die erste und die zweite Operationseinrichtung parallel zu einander vorgesehen sind.

[0030] Gemäß einem weiteren bevorzugten Ausführungsbeispiel der Erfindung ist ferner eine Fehlerkorrekturdekodiereinrichtung ausgebildet, mit einer ersten Operationseinrichtung mit einer Galois-Feldpolynomoperationsschaltung zum Erhalten eines Galois-Feldpolynoms und einer Registerschaltung, die verwendet wird, wenn die Galois-Feldpolynomoperationsschaltung eine Operation durchführt, einer zweiten Operationseinrichtung mit einer Galois-Feldoperationsschaltung zum Durchführen einer Operation unter Verwendung des Galois-Feldpolynoms und einer Universal-Operationsschaltung, einer Registereinrichtung, die verwendet wird, wenn die erste und die zweite Operationseinrichtung Operationen durchführen, und einer Programmerzeu-

gungseinrichtung zur Erzeugung eines Steuerprogramms für die erste und die zweite Operationseinrichtung, wobei die erste und die zweite Operationseinrichtung und die Registereinrichtung parallel zu einander vorgehen sind.

[0031] Weitere Aufgaben, Merkmale und Vorteile der Erfindung werden aus der folgenden Beschreibung in Verbindung mit der beiliegenden Zeichnung ersichtlich.

KURZBESCHREIBUNG DER ZEICHNUNG

[0032] [Fig. 1](#) zeigt ein Ablaufdiagramm eines allgemeinen Euklidischen Algorithmus,

[0033] [Fig. 2](#) zeigt ein Blockschaltbild der Anordnung einer herkömmlichen Fehlerkorrekturdekodiervorrichtung,

[0034] [Fig. 3](#) zeigt ein Blockschaltbild eines Ausführungsbeispiels einer erfindungsgemäßen Fehlerkorrekturdekodiervorrichtung,

[0035] [Fig. 4](#) zeigt ein Ablaufdiagramm eines bei diesem Ausführungsbeispiel verwendeten Euklidischen Algorithmus,

[0036] [Fig. 5](#) zeigt ein Blockschaltbild einer praktischen Konfiguration einer Galois-Feldpolynomoperationsschaltung in [Fig. 3](#) und

[0037] [Fig. 6](#) zeigt ein Blockschaltbild einer praktischen Konfiguration eines Polynomkoeffizientenregisters in [Fig. 3](#).

AUSFÜHRLICHE BESCHREIBUNG DES BEVORZUGTEN AUSFÜHRUNGSBEISPIELS

[0038] Nachstehend wird ein Ausführungsbeispiel der Erfindung unter Bezugnahme auf die beiliegende Zeichnung beschrieben.

[0039] [Fig. 3](#) zeigt ein Blockschaltbild eines Ausführungsbeispiels einer erfindungsgemäßen Fehlerkorrekturdekodiervorrichtung.

[0040] Gemäß [Fig. 3](#) umfasst eine Polynomeinheit (erste Operationseinrichtung) **123**, die das charakteristischste Merkmal dieses Ausführungsbeispiels darstellt, eine Galois-Feldpolynomoperationsschaltung **113** und eine Polynomkoeffizientenregisterschaltung **107** mit einer Chien-Suchschaltung. Diese Polynomeinheit **123** ist mit einem Datenbus parallel zu einer Skalaroperationsschaltung (zweite Operationseinrichtung) **124** verbunden, die eine Skalarregisterschaltung (Registereinrichtung) **114**, eine Universaloperationsschaltung (ALU-Schaltung **116**) und eine Galois-Feldoperationsschaltung (GLU-Schaltung) **118** umfasst.

[0041] Das heißt, die Polynomeinheit **123** ist mit Ausgabebussen (einem A-Bus **108** und einem B-Bus **109**) der Skalarregisterschaltung **114** als Eingangsanschluss für Koeffizientendaten und mit einem Eingabebus (F-Bus) **104** der Skalarregisterschaltung **114** als Eingangs/Ausgangsanschluss für Polynomdaten verbunden. Es wird angenommen, dass ein Galois-Feld $GF(2^8)$ verwendet wird, und jeder Datenbus ein 8-Bit-Bus ist.

[0042] Wie vorstehend beschrieben ist, die Skalarregisterschaltung **114** ein Speicher mit drei Anschlüssen, mit dem der A-Bus **108**, der B-Bus **109** und der F-Bus **106** verbunden sind, und der einen Schreibvorgang und zwei Lesevorgänge unabhängig voneinander ausführen kann.

[0043] Auch die ALU-Schaltung **116** und die GLU-Schaltung **118** in der Skalaroperationsschaltung **124** können Operationen unter Verwendung von Direkteingaben von einem Direktbus (I-Bus) **112** zusätzlich zu Ausgabewerten von der Skalarregisterschaltung **114** durchführen. Das heißt, Auswahlaltungen **115** und **117** wählen den A-Bus **108** oder den I-Bus **112** aus, und für den ausgewählten Wert und den Wert des B-Bus **109** wird eine Operation durchgeführt und das Operationsergebnis wird zu dem F-Bus **104** ausgegeben.

[0044] Daraufhin wird dieses Operationsergebnis in das Skalarregister **114** geschrieben, in die Polynomkoeffizientenregisterschaltung **107** über eine Auswahlaltung **106** oder eine Syndromspeicherschaltung **102** über eine Speicher-IF-Schaltung **103** geschrieben oder zu einer externen Einrichtung (beispielsweise einer Verbindungsschaltung) über eine externe IF-Schaltung **121** ausgegeben.

[0045] Nachstehend wird eine Operation anhand eines Beispiels einer C1-Syndrom-Dekodierung für C1 (80, 72) und C2 (90, 82) Reed-Solomon-Codes als Beispiel beschrieben.

[0046] Eingangsdaten von einem Eingangsanschluss **100** werden in Einheiten von Kodierblöcken (80×90 Bytes) eingegeben. Eine durch eine Galois-Feldadditionsschaltung und eine Galois-Feldkoeffizientenschaltung gebildete Syndromoperationsschaltung **101** berechnet ein C1-Syndrom und ein C2-Syndrom parallel zu einander und schreibt die Berechnungsergebnisse in den Syndromspeicher **102**.

[0047] Der Syndromspeicher **102** hat eine Kapazität zur Speicherung von Syndromen von zwei Blöcken. Werden Syndrome eines Blocks geschrieben, liest gleichzeitig ein Prozessor (der beispielsweise durch die Polynomeinheit **123**, das Skalarregister **114** und die Skalaroperationsschaltung **124** wie vorstehend beschrieben gebildet ist) in den unmittelbar vorhergehenden Block geschriebene Syndrome über die Speicher-IF-Schaltung **103** aus. Auf diese Weise kann eine Fehlerkorrekturdekodierverarbeitung durchgeführt werden.

[0048] Der Prozessor führt die Fehlerkorrekturverarbeitung entsprechend einem Anweisungscode durch, der durch eine Steuerschaltung **119** aus einer Programmspeicherschaltung **120** gelesen wird. Dieser Anweisungscode enthält eine Skalaranweisung und eine Vektoranweisung. Ein Polynomspeicherzugriff und Operationen von Polynomen können durch eine Vektoranweisung durchgeführt werden. Durch Änderung des zu dem Prozessor zuzuführenden Programms kann der Fehlerkorrekturalgorithmus ohne Änderungen von Hardware geändert werden.

[0049] Bei der durch den Prozessor durchgeführten Fehlerkorrekturdekodierverarbeitung werden zuerst ein Fehlerpositionspolynom und ein Fehlerwertpolynom aus einem Syndrompolynom durch einen Euklidischen Algorithmus berechnet. Das heißt, das in die Syndromspeicherschaltung **102** geschriebene C1-Syndrom wird als Syndrompolynom siebten Grades zu der Polynomkoeffizientenregisterschaltung **107** durch die Vektoranweisung übertragen. Auf der Grundlage eines in

[0050] [Fig. 4](#) gezeigten Euklidischen Algorithmus führen die Galois-Feldpolynomoperationsschaltung **113** und die Polynomkoeffizientenregisterschaltung **107** Operationen zum Erhalten des Fehlerpositionspolynoms und des Fehlerwertpolynoms durch.

[0051] In [Fig. 4](#) sind A, B, L und M Galois-Feldpolynome, α und β sind Koeffizienten der Galois-Feldpolynome A und B höchsten Grades, t ist die Fehlerkorrekturleistung, S(z) ist ein Syndrompolynom, $\sigma(z)$ ist ein Fehlerpositionspolynom und $\eta(z)$ ist ein Fehlerwertpolynom.

[0052] [Fig. 5](#) zeigt ein ausführliches Blockschaltbild der Galois-Feldpolynomoperationsschaltung **113**. In der in [Fig. 5](#) gezeigten Konfiguration können bei dem Euklidischen Algorithmus gemäß [Fig. 5](#) verwendete Operationen der folgenden Ausdrücke durch eine einzige Anweisung durchgeführt werden. In den unten stehenden Ausdrücken (14) und (15) sind α und β Galois-Feldelemente und X(z) und Y(z) sind Galois-Feldpolynome.

$$\alpha X(Z) + \beta Y(Z) \quad (14)$$

$$\alpha X(Z) + \beta Y(Z)Z \quad (15)$$

[0053] Wird beispielsweise eine Vektoranweisung entsprechend Ausdruck (14) ausgegeben, werden die Koeffizienten α und β von den Eingangsanschlüssen **200** und **204** eingegeben, die jeweils mit dem A-Bus **108** und dem B-Bus **109** in [Fig. 3](#) verbunden sind, und werden in Koeffizientenregistern **201** und **205** eingestellt. Ferner werden die Galois-Feldpolynome X(Z) und Y(Z) von Eingangsanschlüssen **202** und **206** eingegeben, die mit den Ausgabebussen (einem PA-Bus **110** und einem PB-Bus **111**) des Polynomkoeffizientenregisters **107** verbunden sind.

[0054] Galois-Feldmultiplikationsschaltungen **203** und **208** multiplizieren die eingegebenen Galois-Feldpolynome X(Z) und Y(Z) mit den Koeffizienten α und β , die jeweils in den Koeffizientenregistern **201** und **205** eingestellt sind. Die jeweiligen Produkte werden durch eine Galois-Feldadditionsschaltung **209** addiert und einem Ausgangsanschluss **210** zugeführt.

[0055] Wird andererseits eine Vektoranweisung entsprechend Ausdruck (15) ausgegeben, wird der Grad des vom Eingangsanschluss **206** eingegebenen Polynoms Y(Z) durch eine Gradverschiebungsschaltung **207** um einen Grad verschoben. Danach wird die gleiche Operation wie bei der Vektoranweisung entsprechend dem Ausdruck (14) ausgeführt, und das Operationsergebnis wird dem Ausgangsanschluss **210** zugeführt. Die Aus-

gangsdaten aus dem Ausgangsanschluss **210** werden in das Polynomkoeffizientenregister **107** über einen Bus **105** und die Auswahlhaltung **106** in [Fig. 3](#) geschrieben.

[0056] Bei dem vorstehenden Euklidischen Algorithmus werden die Verarbeitungsschritte wie der Vergleich der Grade und die Erzeugung von Registeradressen, die von Polynomoperationen verschieden sind, parallel zu den Polynomoperationen unter Verwendung der Skalarregistervorrichtung **114** und der Skalaroperationsschaltung **204** durchgeführt.

[0057] Ist die Verarbeitung unter Verwendung des in [Fig. 4](#) gezeigten Euklidischen Algorithmus abgeschlossen, werden das erhaltene Fehlerpositionspolynom $\sigma(Z)$ und das erhaltene Fehlerwertpolynom $\eta(Z)$ an bestimmte Adressen der Polynomkoeffizientenregisterschaltung **107** geschrieben, und die Chien-Suchschaltung berechnet die Wurzel.

[0058] [Fig. 6](#) zeigt ein ausführliches Blockschaltbild der Polynomkoeffizientenregisterschaltung **107**. Diese Polynomkoeffizientenregisterschaltung **107** ist ein Speicher mit drei Anschlüssen ähnlich der Kanalregisterschaltung und kann einen Schreibvorgang von einem Eingangsanschluss **300** über eine Auswahlhaltung **301** und zwei Lesevorgänge für Ausgangsanschlüsse **308** und **309** über eine Auswahlhaltung **307** unabhängig durchführen.

[0059] Adressen zum Auswählen von Registern in dem Polynomkoeffizientenregister **107** umfassen eine obere Adresse zum Auswählen eines Vektors und eine untere Adresse zum Auswählen eines Koeffizienten im Vektor. Diese Adressen werden von einem Eingangsanschluss **306** in eine Steuerschaltung **305** eingegeben. Die Steuerschaltung **305** dekodiert die eingegebenen Adressen und wählt Schreib- und Leseregister durch Steuerung der Auswahlhaltungen **301** und **307** aus.

[0060] Als Beispiel wird angenommen, dass nachstehend gezeigte eingegebene Adressen zum Auswählen einer Universalregisterschaltung **302** oder einer Chien-Suchregisterschaltung **303** oder einer Steuerregisterschaltung **304** und zur Auswahl interner Register der Schaltung verwendet werden.

Obere Adressen	Untere Adressen	
0 - 4	0 - 8	Universalregister
5 - 6	0 - 7	Chien- Suchregister
7	0	Steuerregister

[0061] Der höchste Grad des in der Fehlerkorrekturverarbeitung verwendeten Polynoms ist Z^8 . Dementsprechend kann in der Universalregisterschaltung **302** auf Register für fünf Vektoren beliebig zugegriffen werden unter der Annahme, dass ein 9-Byte-Koeffizient einen Vektor bildet, d. h. auf 9×5 Byte-Register **3100** bis **3148**. Bei dieser Konfiguration sind die Register **3100** bis **3118** an oberen Adressen 0 und 1 als Register zugeordnet, die bei Polynomoperationen verwendet werden, die durch den Euklidischen Algorithmus durchgeführt werden. Die Register **3120** bis **3148** an oberen Adressen 2 bis 4 sind als Register zur Speicherung der Ergebnisse der Chien-Suche zugewiesen.

[0062] Auch in der Chien-Suchschaltung **303** kann auf 8×2 Byte-Register **3200** bis **3207** und **3240** bis **3247** beliebig zugegriffen werden. Diese Register sind mit Koeffizienteneinheiten **3210** bis **3217** und **3250** bis **3257** zum Multiplizieren von Galois-Feldkoeffizienten α^{259} bis α^{247} verbunden.

[0063] Ausgaben aus den Galois-Feldkoeffizienteneinheiten **3210** bis **3217** werden einer Galois-Feldadditionsschaltung **3230** zugeführt, wo eine Galois-Feldaddition für alle Koeffizienten durchgeführt wird. Die Summe kann über die Auswahlhaltung **301** in ein anderes Register geschrieben werden. Ferner werden Ausgaben aus den Galois-Feldkoeffizienteneinheiten **3250** bis **3257** einer Galois-Feldadditionsschaltung **3260** zugeführt, wo eine Galois-Feldaddition für alle Koeffizienten durchgeführt wird, und die Summe wird zu der Auswahlhaltung **301** ausgegeben. Des Weiteren werden Ausgaben hinsichtlich geradzahligter Grade aus den Galois-Feldkoeffizienteneinheiten **3250** bis **3257** einer Galois-Feldadditionsschaltung **3260** zugeführt, wo eine Galois-Feldaddition für diese Koeffizienten geradzahligem Grades durchgeführt wird, und die Summe wird zu der Auswahlhaltung **301** ausgegeben. Des Weiteren werden Ausgaben hinsichtlich geradzahligter Grade aus den Galois-Feldkoeffizienteneinheiten **3250** bis **3257** einer Galois-Feldadditionsschaltung **3** zugeführt, wo eine Galois-Feldaddition für diese Koeffizienten geradzahligem Grades durchgeführt wird, und die Summe wird zu

der Auswahlschaltung **301** ausgegeben.

[0064] Die Ausgabe aus der Galois-Feldadditionsschaltung **3260** wird auch einer Nullerfassungsschaltung **3261** zugeführt. Die Nullerfassungsschaltung **3261** überprüft, ob der Ausgabewert aus der Galois-Feldadditionsschaltung **3260** „0“ ist. Ist der ausgegebene Wert „0“, führt die Nullerfassungsschaltung **3261** der Steuerungsschaltung **305** und einer Fehleradressenzählschaltung **3262** ein Steuersignal zu. Entsprechend dem Steuersignal von der Nullerfassungsschaltung **3261** gibt die Fehleradressenzählschaltung **3262** eine Fehleradresse zu der Auswahlschaltung **301** aus.

[0065] Der Betrieb der Polynomkoeffizientenregisterschaltung **107** mit dem in [Fig. 6](#) gezeigten Aufbau wird nachstehend beschrieben. Zuerst werden das Fehlerwertpolynom $\eta(Z)$ und das Fehlerpositionspolynom $\sigma(Z)$, die durch die Galois-Feldpolynomoperationsschaltung **113** in [Fig. 3](#) unter Verwendung des Euklidischen Algorithmus in [Fig. 4](#) berechnet werden, jeweils in die Chien-Suchregister **320** bis **3207** und **3240** bis **3247** an oberen Adressen 5 und 6 geschrieben. Nachdem das Fehlerpositionspolynom $\sigma(Z)$ und das Fehlerwertpolynom $\eta(Z)$ so in diesen Registern eingestellt wurden, wird ein Steuerregister **330** an der oberen Adresse 7 zum Einstellen einer Codelänge (hier 80) ausgewählt. Gleichzeitig wird ein Flag eines Chien-Suchstarts gesetzt.

[0066] Ist das Chien-Suchstartflag gesetzt, gibt die Steuerungsschaltung **305** die I/O-Anschlüsse der Chien-Suchregister **3200** bis **3207** und **3240** bis **3247** durch Steuern der Auswahlschaltungen **301** und **307** frei, und steuert diese Register zum sequentiellen Einsetzen der Potenzen von α in das Fehlerwertpolynom $\eta(Z)$ und das Fehlerpositionspolynom $\sigma(Z)$, bis die Codelänge erreicht ist. Gleichzeitig führt die Steuerungsschaltung **305** die Ausgabe aus der Galois-Feldadditionsschaltung **3260** der Nullerfassungsschaltung **3261** zu, die überprüft, ob $\sigma(\alpha^1) = 0$ ist.

[0067] Erfasst die Nullerfassungsschaltung **3261**, dass $\sigma(\alpha^1) = 0$ ist, schreibt die Steuerungsschaltung **305** über die Auswahlschaltung **301** die Ausgabe aus der Galois-Feldadditionsschaltung **3260** als $\eta(\alpha^1)$ in das Register an der oberen Adresse 2, die Ausgabe aus der Galois-Feldadditionsschaltung **3263** als formelle Ableitung $\sigma'(\alpha^1)$ von $\sigma(\alpha^1)$ in das Register an der oberen Adresse 3, und die Ausgabe aus dem Fehleradressenzähler **3262** als Fehleradresse i in das Register an der oberen Adresse 4.

[0068] Danach ist die Chien-Suche abgeschlossen, wenn die Potenzen von α 80-Mal entsprechend der Codelänge eingesetzt wurden. Ist die Chien-Suche abgeschlossen, schreibt die Steuerungsschaltung **305** ein Chien-Suche-Ende-Flag in das Steuerregister **3300**. Das Ende der Chien-Suche kann durch Auslesen des Flags aus dem Steuerregister **3300** durch das Programm erkannt werden.

[0069] Während die Chien-Suchschaltung in der Polynomkoeffizientenregisterschaltung **107** die Chien-Suchverarbeitung durchführt, kann die Galois-Feldpolynomoperationsschaltung **113** gleichzeitig den Euklidischen Algorithmus für den nächsten Code unter Verwendung der Register an oberen Adressen 0 und 1 ausführen; es ist nicht erforderlich, die Verarbeitung zur Durchführung der Chien-Suche anzuhalten. Dementsprechend kann die Fehlerkorrekturverarbeitung mit hoher Geschwindigkeit durchgeführt werden.

[0070] Ist die Chien-Suchverarbeitung abgeschlossen, kann auch ein Fehlerwert e_i durch Ausführen von unten stehender Gleichung (16) unter Verwendung der GLU-Schaltung **118** in der Skalaroperationsschaltung **124** berechnet werden.

$$e_i = \frac{\eta(\alpha^i)}{\sigma'(\alpha^i)} \quad (16)$$

[0071] Die durch die Chien-Suchschaltung in der Polynomkoeffizientenregisterschaltung **107** berechnete Fehleradresse i und der durch die GLU-Schaltung **118** berechnete Fehlerwert e_i werden einer Korrekturschaltung (nicht gezeigt) von einem Ausgangsanschluss **122** über die Speicher-IF-Schaltung **103** und die externe IF-Schaltung **121** zugeführt, und eine Fehlerkorrektur wird durchgeführt. Außerdem wird das C2-Syndrom in der Syndromschaltung **102** neu geschrieben. Die C1-Korrekturverarbeitung kann durch Wiederholen der vorstehenden Prozedur mit hoher Geschwindigkeit ausgeführt werden.

[0072] Eine Löschungskorrektur kann durch die Durchführung einer Fehlerkorrekturverarbeitung unter der Annahme realisiert werden, dass die Fehlerposition bereits gefunden ist.

[0073] Eine Codierverarbeitung wird auch durch die Durchführung einer Löschungskorrektur für die Paritätsposition bewirkt.

[0074] Wie vorstehend beschrieben weist die Fehlerkorrekturdekodier Vorrichtung dieses Ausführungsbeispiels eine Prozessorkonfiguration auf, bei der eine erste und eine zweite Operationseinrichtung parallel zueinander vorgesehen sind, und entsprechend durch eine Programmierungseinrichtung erzeugte Programme gesteuert werden. Daher können alle Änderungen im Algorithmus zur Durchführung einer Fehlerkorrektur mittels Software realisiert werden, wodurch ein äußerst flexibles System mit geringem Schaltungsumfang realisiert werden kann. Obwohl in der zweiten Operationseinrichtung eine Galois-Feldoperationsschaltung eine Operation durchführt, kann eine Galois-Feldpolynomoperationsschaltung in der zu der zweiten Operationseinrichtung parallel vorgesehenen ersten Operationseinrichtung ein Galois-Feldpolynom für die nächsten codierten Daten berechnen. Dementsprechend kann die Verarbeitungsgeschwindigkeit erhöht werden.

[0075] Ferner sind Galois-Feldkoeffizienteneinheiten und Galois-Feldadditionsschaltungen zu einigen eine Registerschaltung bildenden Registern hinzugefügt, und Wurzeln werden durch eine Chien-Suche für in diese Register geschriebene Galois-Feldpolynome berechnet. Da somit eine Chien-Suchfunktion für einen Teil der Registerschaltung vorgesehen ist, können durch eine Galois-Feldpolynomoperationsschaltung durchgeführte Operationen für Galois-Feldpolynome und Berechnungen von Fehlerpositionen durch eine Chien-Suche parallel zueinander durch die Steuerung von Programmen ausgeführt werden. Dies erhöht weiter die Verarbeitungseffizienz und realisiert eine Fehlerkorrekturverarbeitung mit hoher Geschwindigkeit.

[0076] Die vorstehende Beschreibung dient also lediglich der Veranschaulichung der Erfindung und soll keine Einschränkung darstellen.

[0077] Der Schutzbereich der Erfindung ist daher lediglich durch die beiliegenden Patentansprüche bestimmt.

Patentansprüche

1. Fehlerkorrekturvorrichtung mit einer Galois-Feld-Polynomerzeugungsschaltung (**113**) zur Erzeugung eines Fehlerpositionspolynoms und eines Fehlerwertpolynoms, einer Chien-Suchschaltung (**107**) zur Durchführung einer Chien-Suche, einer Berechnungsschaltung (**118**) zur Berechnung von Fehlerwerten und einer Steuerschaltung (**119**) zur Steuerung der Galois-Feld-Polynomerzeugungsschaltung, der Chien-Suchschaltung und der Berechnungsschaltung, wobei die Galois-Feld-Polynomerzeugungsschaltung und die Chien-Suchschaltung in einer ersten Operationsschaltung (**123**) integriert vorgesehen sind, wobei die erste Operationsschaltung derart eingerichtet ist, dass die Galois-Feld-Polynomerzeugungsschaltung zur Durchführung eines nächsten Prozesses eingerichtet ist, während die Chien-Suchschaltung die Chien-Suche durchführt, wobei die Berechnungsschaltung in einer zweiten Operationsschaltung (**124**) vorgesehen ist, die parallel zu der ersten Operationsschaltung vorgesehen ist, und wobei die Steuerschaltung zur Steuerung der ersten und zweiten Operationsschaltung unter Verwendung eines in einem Speicher (**120**) gespeicherten Steuerprogramms eingerichtet ist.

2. Vorrichtung nach Anspruch 1, wobei die Galois-Feld-Polynomerzeugungsschaltung zur Erzeugung des Fehlerpositionspolynoms und des Fehlerwertpolynoms unter Verwendung eines euklidischen Algorithmus eingerichtet ist.

3. Vorrichtung nach Anspruch 1 oder 2, wobei die Galois-Feld-Polynomerzeugungsschaltung zur Durchführung einer durch die folgenden Ausdrücke definierten Operation eingerichtet ist:

$$aX(Z) + \beta Y(Z)$$

$$aX(Z) + \beta Y(Z)Z$$

wobei α und β Galois-Feld-Elemente und $X(Z)$ und $Y(Z)$ Galois-Feld-Polynome sind.

4. Verfahren zum Betreiben einer Fehlerkorrekturvorrichtung, mit Betreiben einer Galois-Feld-Polynomerzeugungsschaltung (**113**) zur Erzeugung eines Fehlerpositionspolynoms und eines Fehlerwertpolynoms, Betreiben einer Chien-Suchschaltung (**107**) zur Durchführung einer Chien-Suche, Betreiben einer Berechnungsschaltung (**124**) zur Berechnung von Fehlerwerten und Betreiben einer Steuerschaltung (**119**) zur Steuerung der Galois-Feld-Polynomerzeugungsschaltung, der Chi-

en-Suchschaltung und der Berechnungsschaltung,
 wobei die Galois-Feld-Polynomerzeugungsschaltung und die Chien-Suchschaltung in einer ersten Operationsschaltung (**123**) integriert vorgesehen sind, die erste Operationsschaltung derart eingerichtet ist, dass die Galois-Feld-Polynomerzeugungsschaltung zur Durchführung eines nächsten Prozesses betreibbar ist, während die Chien-Suchschaltung die Chien-Suche durchführt,
 wobei die Berechnungsschaltung in einer zweiten Operationsschaltung (**124**) vorgesehen ist, die parallel zur ersten Operationsschaltung vorgesehen ist, und
 wobei die Steuerschaltung zur Steuerung der ersten und zweiten Operationsschaltung unter Verwendung eines in einem Speicher (**120**) gespeicherten Steuerprogramms betreibbar ist.

5. Verfahren nach Anspruch 4, wobei die Galois-Feld-Polynomerzeugungsschaltung zur Erzeugung des Fehlerpositionspolynoms und des Fehlerwertpolynoms unter Verwendung eines euklidischen Algorithmus betreibbar ist.

6. Verfahren nach Anspruch 4 oder 5, wobei die Galois-Feld-Polynomerzeugungsschaltung zur Durchführung einer durch die folgenden Ausdrücke definierten Operation betreibbar ist:

$$\alpha X(Z) + \beta Y(Z)$$

$$\alpha X(Z) + \beta Y(Z)Z$$

wobei α und β Galois-Feld-Elemente und $X(Z)$ und $Y(Z)$ Galois-Feld-Polynome sind.

Es folgen 6 Blatt Zeichnungen

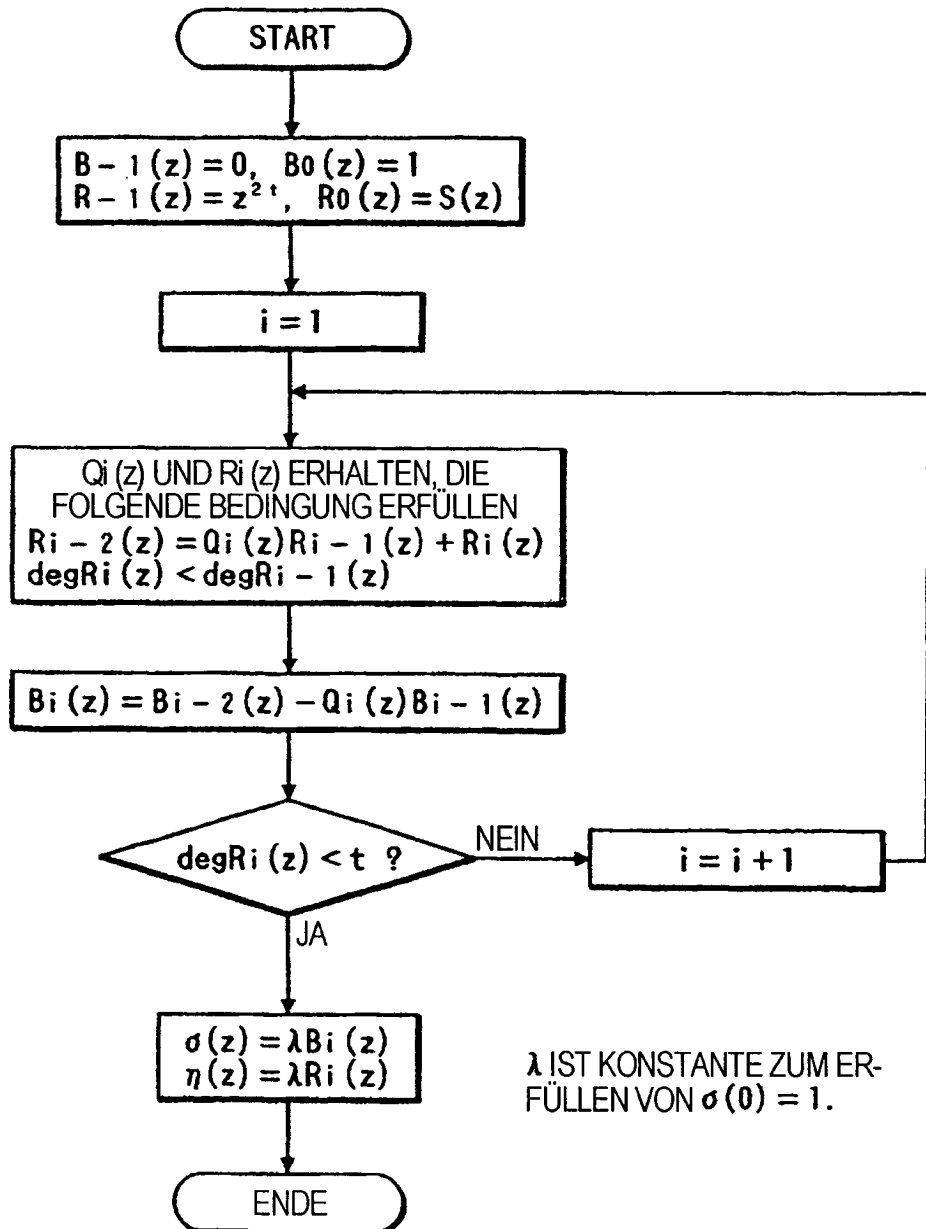
FIG. 1

FIG. 2

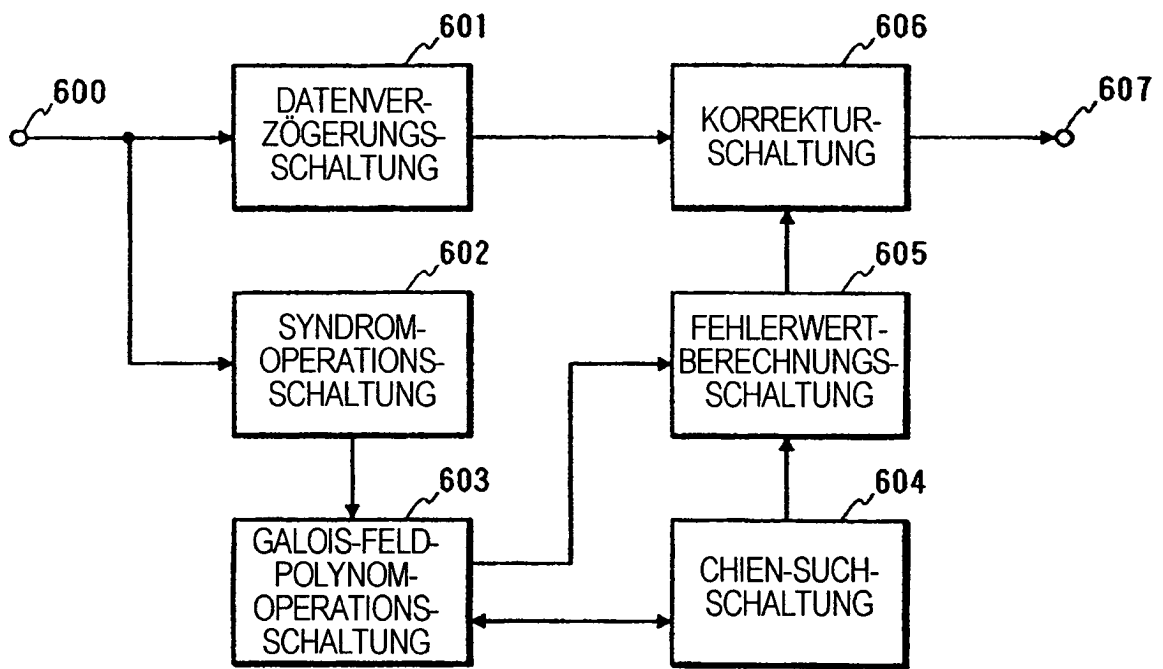


FIG. 3

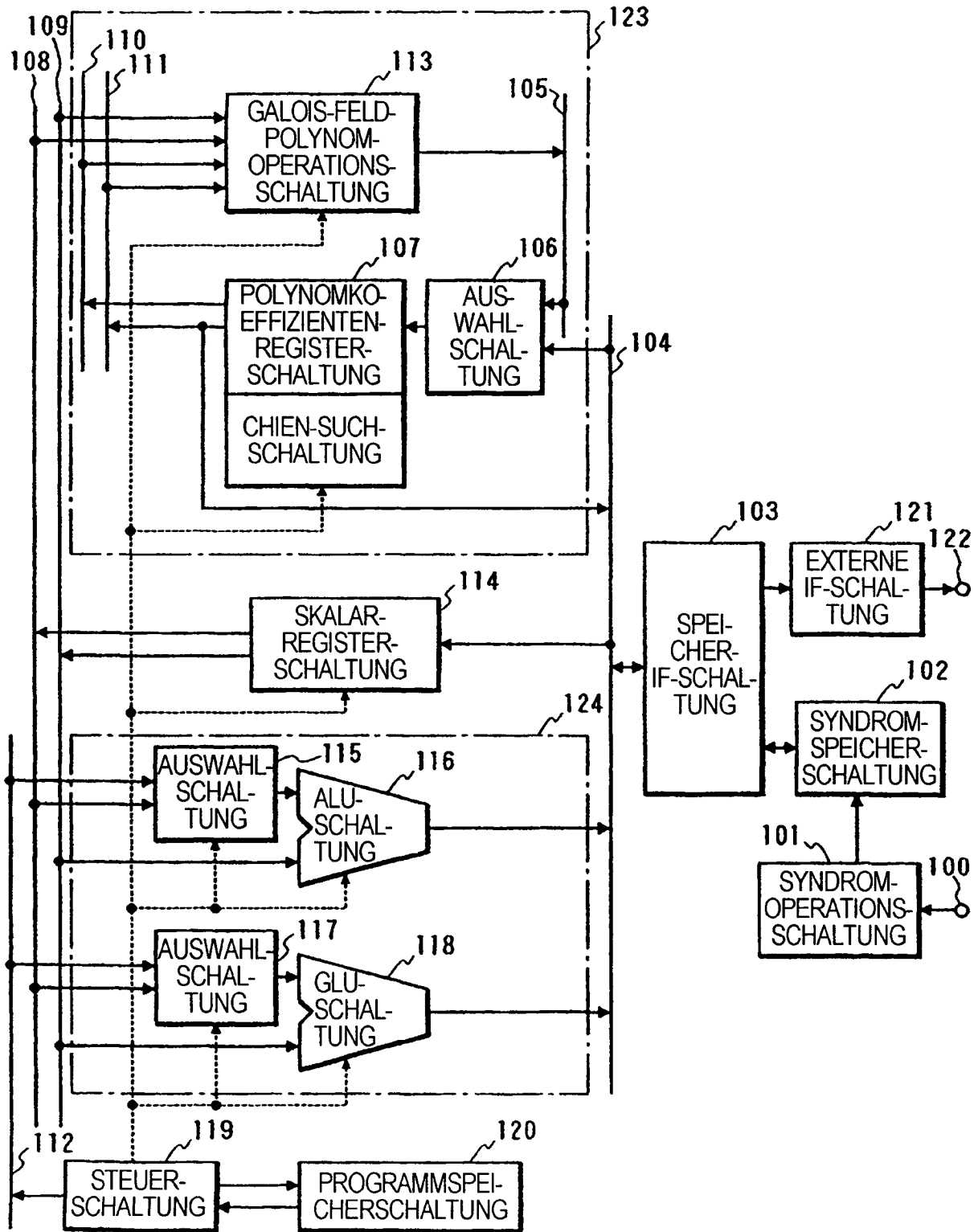


FIG. 4

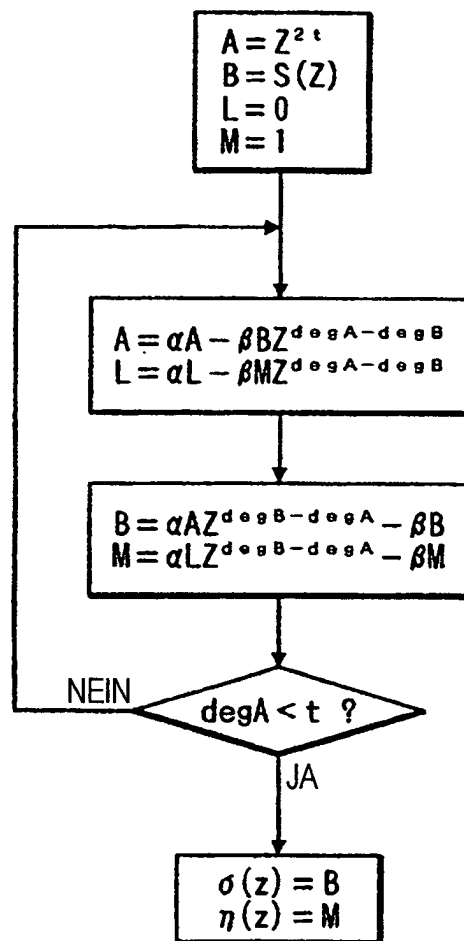


FIG. 5

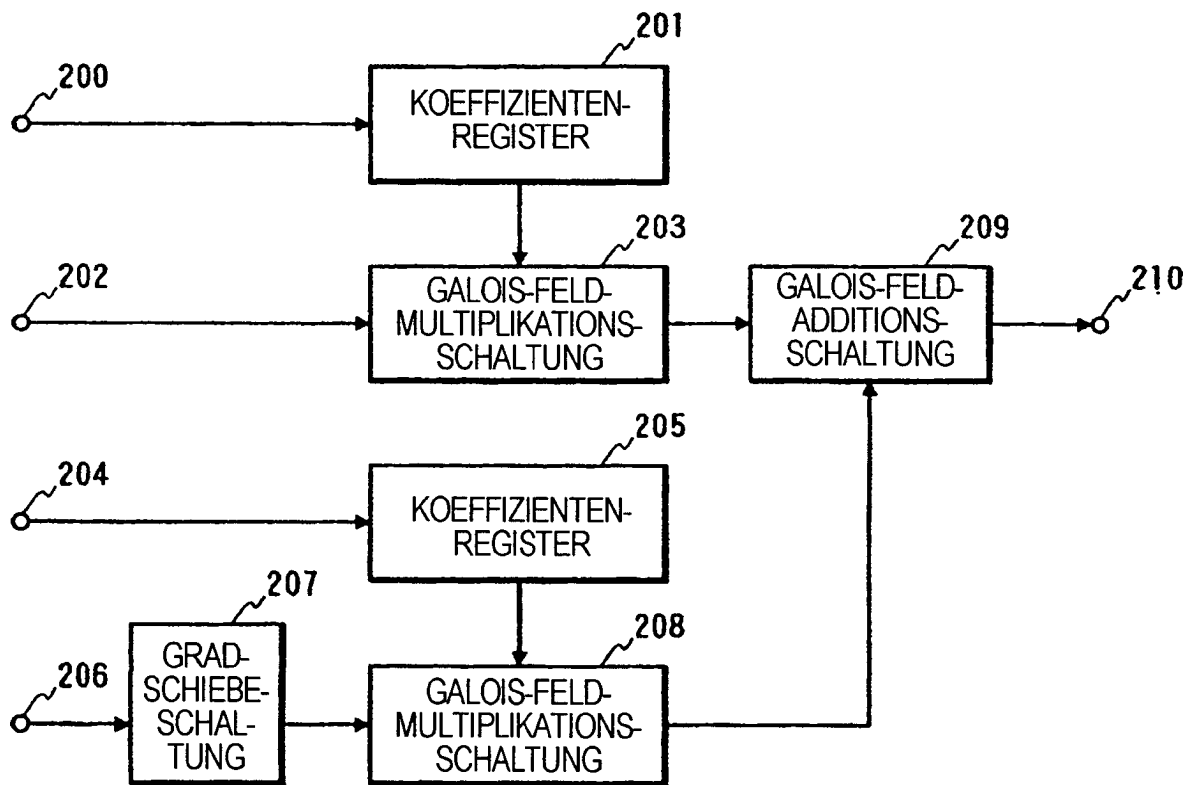


FIG. 6

